R75-34
MAY 1975

NASA CR-
141792

FINAL REPORT

SOURCE ENCODING FOR ORBITER COMMUNICATIONS LINKS

Prepared for
NASA JOHNSON SPACE CENTER
Houston, Texas 77058

(NASA-CR-141792) SOURCE ENCODING FOR ORBITER COMMUNICATIONS LINKS Final Report (Delco Electronics, Santa Barbara, Calif.) 116 p HC $5.25 CSCL 17B

N75-23653

Unclas
G3/17 21629

In response to
Contract No. NAS9-14402

Delco Electronics

General Motors Corporation
- Santa Barbara Operations
Santa Barbara, California

R75-34
MAY 1975

FINAL REPORT

SOURCE ENCODING FOR ORBITER COMMUNICATIONS LINKS

Prepared for
NASA JOHNSON SPACE CENTER
Houston, Texas 77058

In response to
Contract No. NAS9-14402

Delco Electronics

General Motors Corporation
- Santa Barbara Operations
Santa Barbara, California

TABLE OF CONTENTS (Sheet 1 of 2)

TABLE OF CONTENTS (Sheet 2 of 2)

LIST OF ILLUSTRATIONS

LIST OF TABLES

1.0 INTRODUCTION

The baseline communications link for the Space Shuttle involves two way transmission of digital information via a relay satellite system. These links as presently planned appear to be marginal. The purpose of this study was to evaluate the feasibility of using data compression to improve link efficiency as an alternative to increased transmitter power, reducing receiver noise figures, increasing antenna gain through more stringent Orbiter attitude constrains, etc.

Delco Electronics has evolved a method of encoding digital data which permits low bandwidth encoding as well as a unique system of adaptive run length encoding. The purpose of this study was to evaluate the effectivity of these techniques for the air-to-ground link and for the bandwidth-limited ground-to-ground data link used for the Orbiter downlink data.

2.0 SYNOPSIS OF TASKS

2.1 TASK 3.2.1 REDUNDANCY REMOVAL, ORBITER-TO-GROUND LINK

2.1.1 Purpose and Scope

The purpose of this task is to establish the feasibility of improving the overall link performance by using data compression to reduce the transmitted bit rate. Conventional methods for data compression require that a substantial number of overhead bits be added to the transmitted data; for time division multiplexed telemetry data, the overhead management is such that negative throughput gains are frequently produced. The Delco system provides an overhead management technique which assures that all overhead bits added to the transmitted data result in a net decrease in the quantity of bits transmitted.

The baseband Delco data, however, is not conventional in character; the baseband signal is a two-voltage, digital waveform composed of transitions between voltage levels in increments of 1.5 to 4.5 bit times in one-half bit intervals. Within this family of pulses, there exist several which are not used to encode serial digital source data. These "unique characters" are then used to signal both the existence of and the quantity of redundancy. To evaluate the effectivity of the data compression system it is necessary to determine the performance of the Delco data in noisy channels.

The primary thrust of this task is, therefore:

- Compare the bit error rate versus signal-to-noise ratio of the Delco system with that of a well known modulation technique to determine the compression required
- Determine the throughput obtainable after compression as a function of redundancy content
- Estimate the redundancy content of the downlink telemetry data
- Compare the compression ratio required with the throughput obtainable to determine feasibility
- Generate a preliminary mechanization incorporating the compression system in the air-to-ground link.

2.1.2 Significant Results Obtained

2.1.2.1 Bit Error Rate Comparison

In order to compare the Delco system with (optimum) systems employing conventional modulation techniques, optimum Delco Level I systems have been identified and an analysis of their performance carried out. This subtask has been accomplished and is reported in Appendix B. The study has been restricted to baseband systems, but comparisons among baseband systems generally carry over to the corresponding modulation systems. (See Reference 1 of Appendix B.) Optimum Level I systems have been identified and studied for both the usual Level I application in which there is a bandwidth constraint and for applications that have no significant bandwidth restrictions. These systems are not truly optimum in the sense that their performance can be improved by utilizing sequential decoding techniques due to the correlation in the Level I bit streams. However, sequential detection of Level I has not been attempted in the laboratory, and its analysis was considered outside the scope of the study. Sequential detection of Level I will improve the error performance at the cost of some increase in hardware complexity.

The basic results of the study are shown in a simplified form in Table 1. For more detailed results and discussion, see Figure 23 and Section 4.4 of Appendix B.

MODULATION TECHNIQUE	SNR TO ACHIEVE A 10^{-4} BER
Conventional	8.4 dB
Conventional with Coding Gain	3.8 dB
Level I	11.7 dB
Level I (Bandwidth Restricted)	13.5 dB

Table 1. Bit Error Rate Comparison

The signal-to-noise ratio (SNR) is defined to be the ratio of the received signal power to the noise power in a bandwidth equal to the information rate. The entry labeled "Conventional with Coding Gain" is extracted from the article by Batson and Moorehead ("A Digital Communications System for Manned Spaceflight Application," B. H. Batson and R.W. Moorehead) in which a 4.6 dB reduction in required signal-to-noise ratio is shown with the use of convolutional encoding and Viterbi decoding.

Table 2 shows the reduction in signal-to-noise ratio requirements that would be realized through the use of Level II and Level III as a function of the compression ratio obtained.

Compression Ratio	1:1	2.1:1	6.2:1	12.3:1
Required SNR (before compression to achieve 11.7 dB SNR after compression)	11.7 dB	8.4 dB	3.8 dB	0.8 dB

Table 2. SNR to Achieve 10^{-4} BER Versus Compression Ratio

This table has been constructed from the equation (SNR reduction) = 10 log (compression ratio) and, of course, is valid regardless of the means used to obtain the compression and thus reduce the data rate. The entries were chosen to show that a compression of 2.1:1 suffices to bring Level I up to the performance of the conventional system; a compression of 6.2:1 achieves the error rate of the conventional system with convolutional-Viterbi; and a compression of 12.3:1 reduces the SNR requirements to 3 dB below those of coded conventional.

2.1.2.2 Throughput Improvement

The compression system considered is one in which the sampled signal value is sent during selected master PCM frames or when the signal values differ from a previous sample by a predetermined amount. When a signal or string of signals are not sufficiently different from previously transmitted values, uniquely coded waveforms are substituted for these signals. These waveforms occupy significantly less space in the communications channel and permit a throughput improvement. Buffering of the information to be transmitted provides for the generation of a constant bit rate PCM serial output, which carries all information contained in the original PCM data, but at a much lower bit rate. The reduction in bit rate achievable is related to the redundancy contained in the source data and to the overhead required to identify the data channels which were not sent in original signal form.

This system generates no overhead during nonredundant periods and from 4 to 23 bits for redundant strings of from 1 to 256 data channels. The compression produced varies from 1:1 for zero redundancy to over 90:1 for string lengths of approximately 250. The system adaptively selects the preferred method for encoding the output PCM data – for nonredundant channels, the data are encoded in Level I; for redundant strings of up to eight bits, the data are encoded by Level II; for strings of nine bits or more, the data are encoded by Level III coding. The mode of operation selected is easily detected by the receiving system due to the difference in the use of the unique characters used to encode the redundancy.

The differences between the encoding levels can be seen from examination of Figures 1 through 4. The rules for encoding data in Level I are portrayed in Figure 1. In this mode of operation, the unique character is used to encode three consecutive encoded zeros; this eliminates the possibility of generating large, low frequency components in the baseband Level I signal. When the unique character is used for this purpose, it is injected as near the preceeding transition as is permitted by the encoding rules. It is never delayed from this position when used to encode data by Level I.

Figure 2 illustrates how this character is delayed to produce Levels II and III; the two following figures are examples of that encoding. The Level III mode of operation is indicated by two sequential, unique characters, each with a one bit injection delay. The run length counter is a binary count of the run length, transmitted most significant bit first. The counter value is always followed by a bit controlled to a logic ONE state and a delayed unique character. This arrangement clearly indicates the presence of Level III encoding, and the counter length is indicated by the terminator coding.

Figure 5 illustrates the relationship between string length and overhead and the achieveable compression.

2.1.2.3 Downlink Redundancy Content

In order to arrive at an achievable throughput improvement for Orbiter data, a sample PCM format was constructed using selected Titan data signals in expanded quantities. A 360-word format was synthesized to establish the serial stringing produced by the redundancy contained in the sampled signal values. This format is contained in Appendix A.

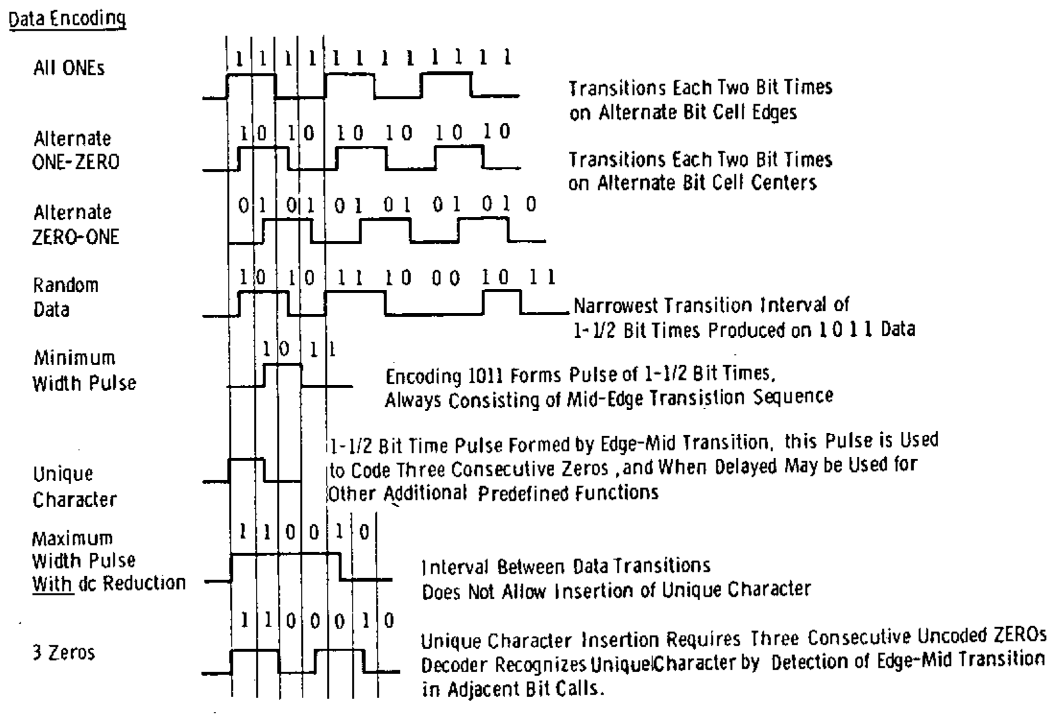


Figure 1. Level I Encoding

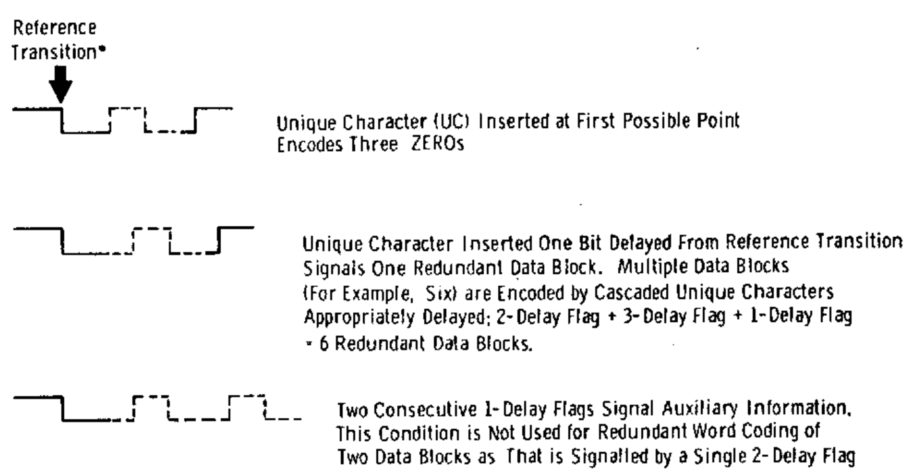


Figure 2. Basis for Level II and III Systems

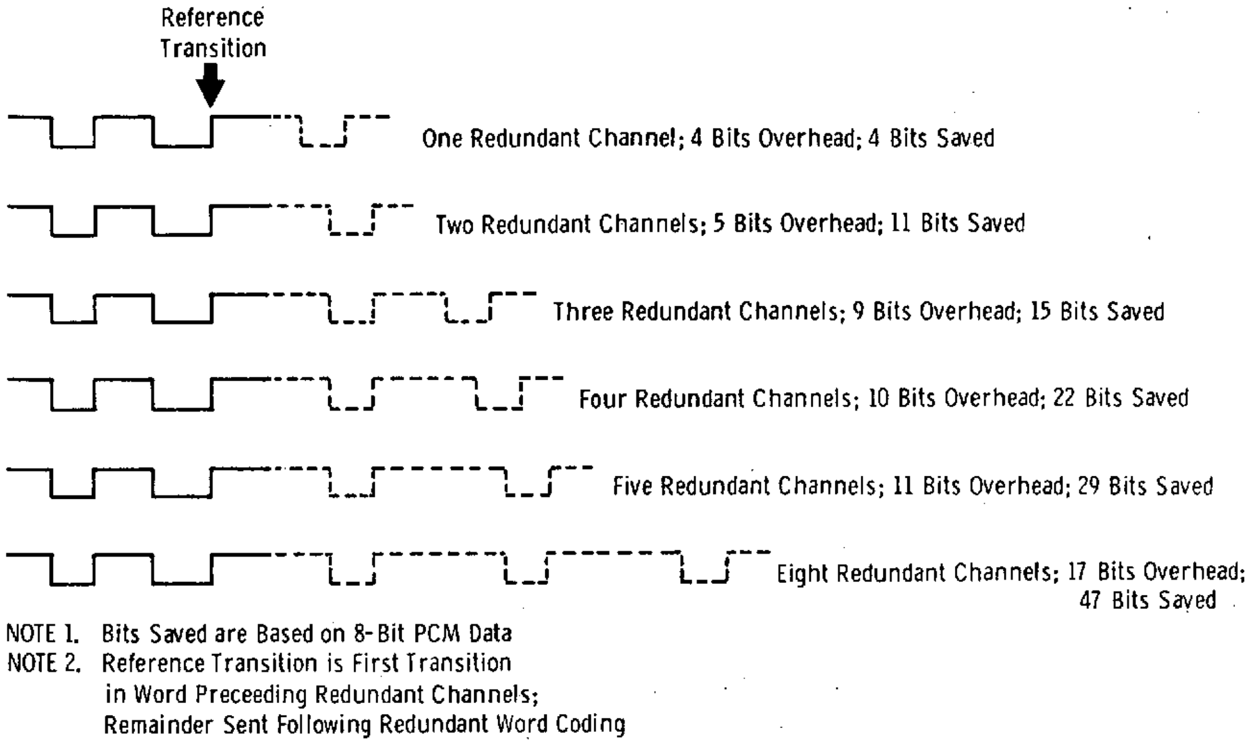


Figure 3. Level II Encoding

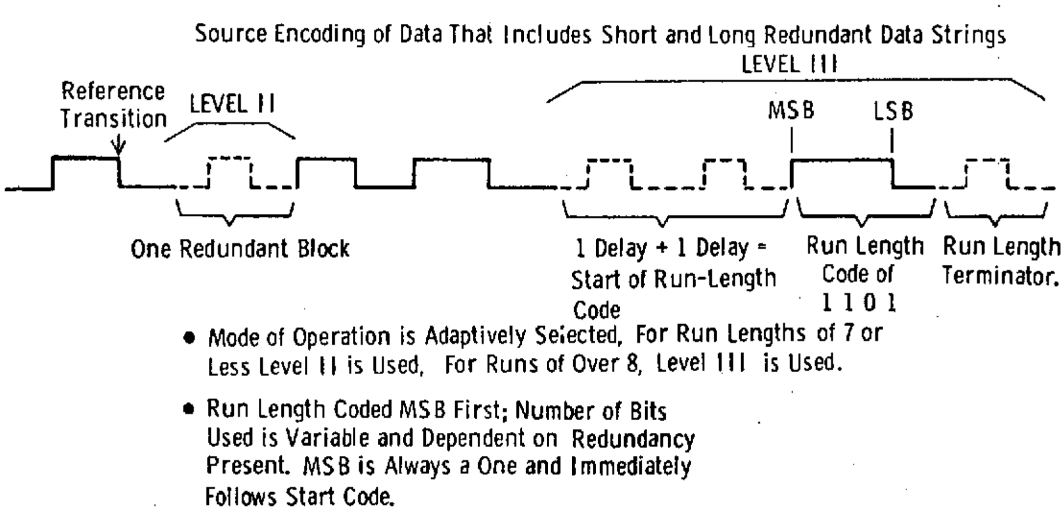


Figure 4. Level III Encoding

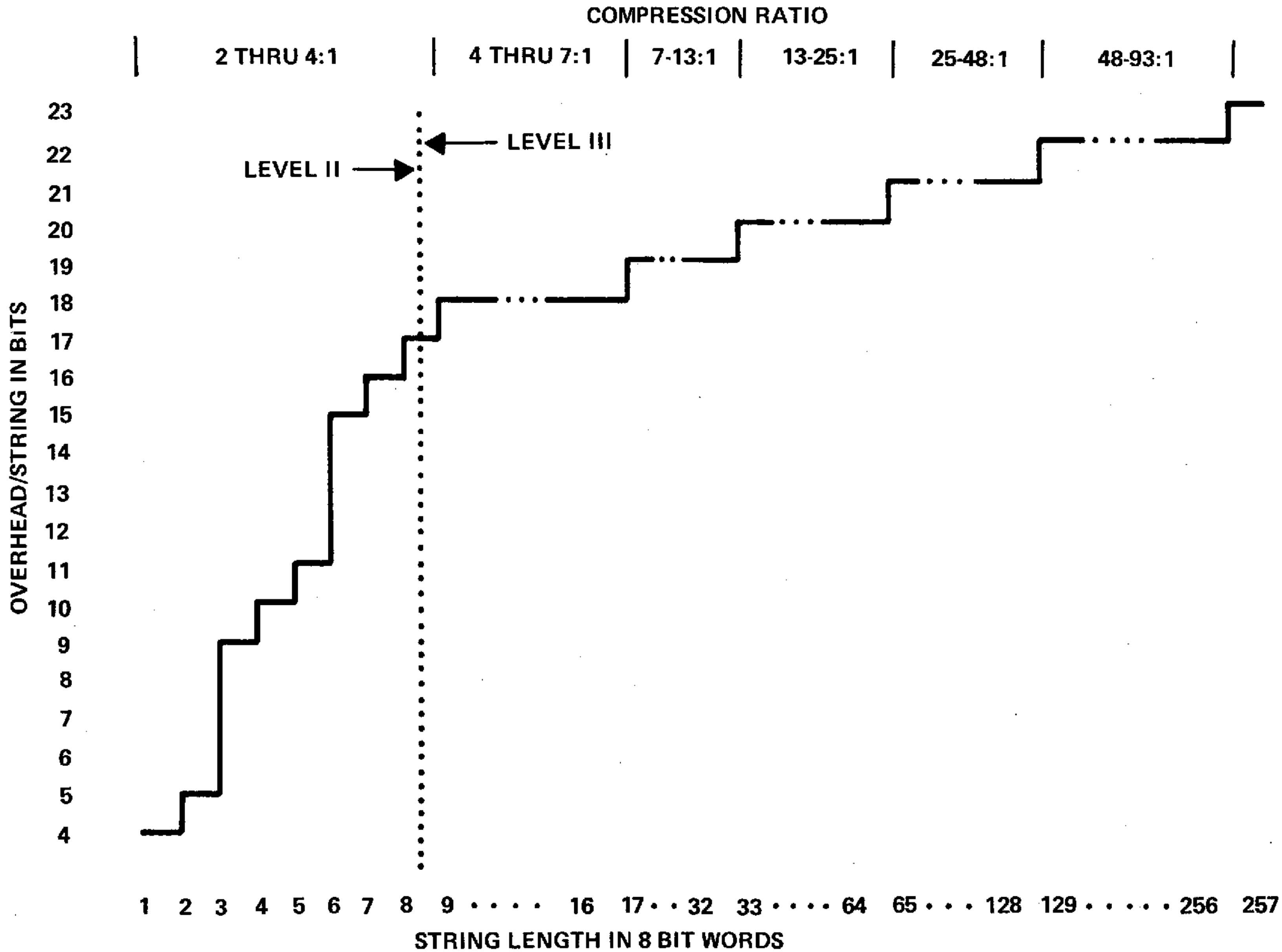


Figure 5. String Length/Overhead Relationships

These signals were examined during the boost and coast phases, the signal activity during the coast period was selected as being most representative of Orbiter on-orbit data conditions where the weak link conditions exist. During this period the signals are highly redundant, and fewer than two data channels require transmission out of each frame of data. A throughput advantage of over 100:1 is suggested; however, periodic signal updates, synchronization codes, and a reasonable safety factor reduce this factor significantly. A more reasonable design goal is 10 to 20:1, which permits a greater variation to exist due to signal anomalies without overloading the compression system. The redundancy present in the boost phase, where maximum signal activity exists, permits compression of approximately 10:1. The string length analysis for those flight phases are also contained in Appendix A.

2.1.2.4 Feasibility

The results of Section 2.1.2.1 above show that if Levels I, II, and III can develop a compression of 10:1 or better, the signal-to-noise requirements will be reduced substantially compared with a conventional system using convolutional encoding – Viterbi decoding. Section 2.1.2.3 conservatively estimates that compression ratios of 10 to 20:1 are achieveable with the downlink data. It is therefore concluded that not only is the application of Delco's source encoding techniques to the Orbiter-to-ground communications links feasible, but that the use of Levels I, II, and III provide a sufficient reduction in transmitted bit rate so as to negate the requirement for channel encoding.

2.1.2.5 Mechanization

A generalized data compression system is shown in Figures 6 and 7. This system accumulates data from the existing PCM Master Unit (PCMMU). Each data channel is compared with its previous value, and channels which contain new information are tagged for transmission. The buffered data are searched for these tags in an order identical to the original PCM format. All channels tagged are included in the output data stream. It is apparent that a variable word/frame rate exists for the output data. The variable word/frame rate is restored by buffering in the ground system for data playback in conventional data systems, if required.

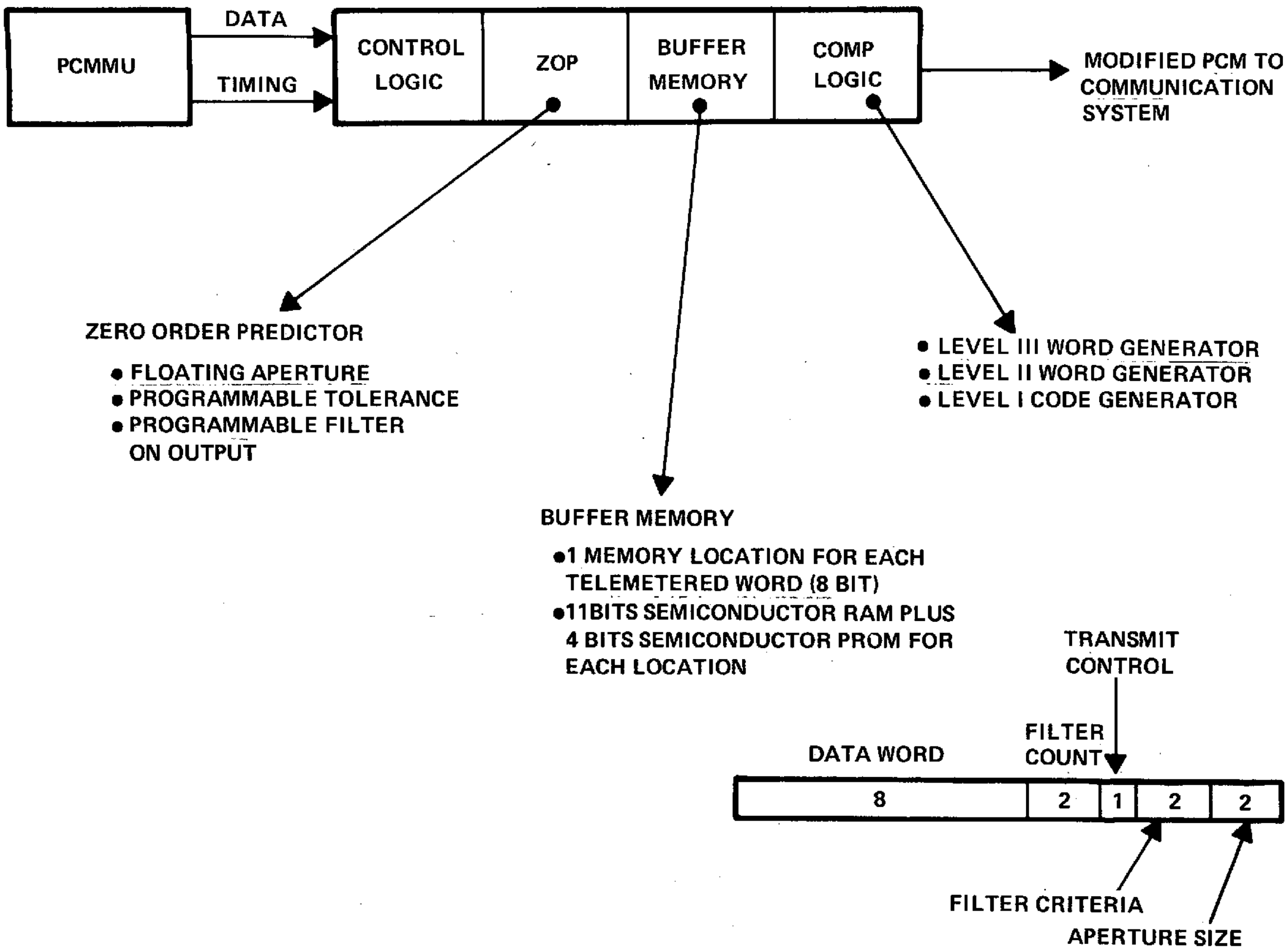


Figure 6. Airborne Compressor Functional Diagram

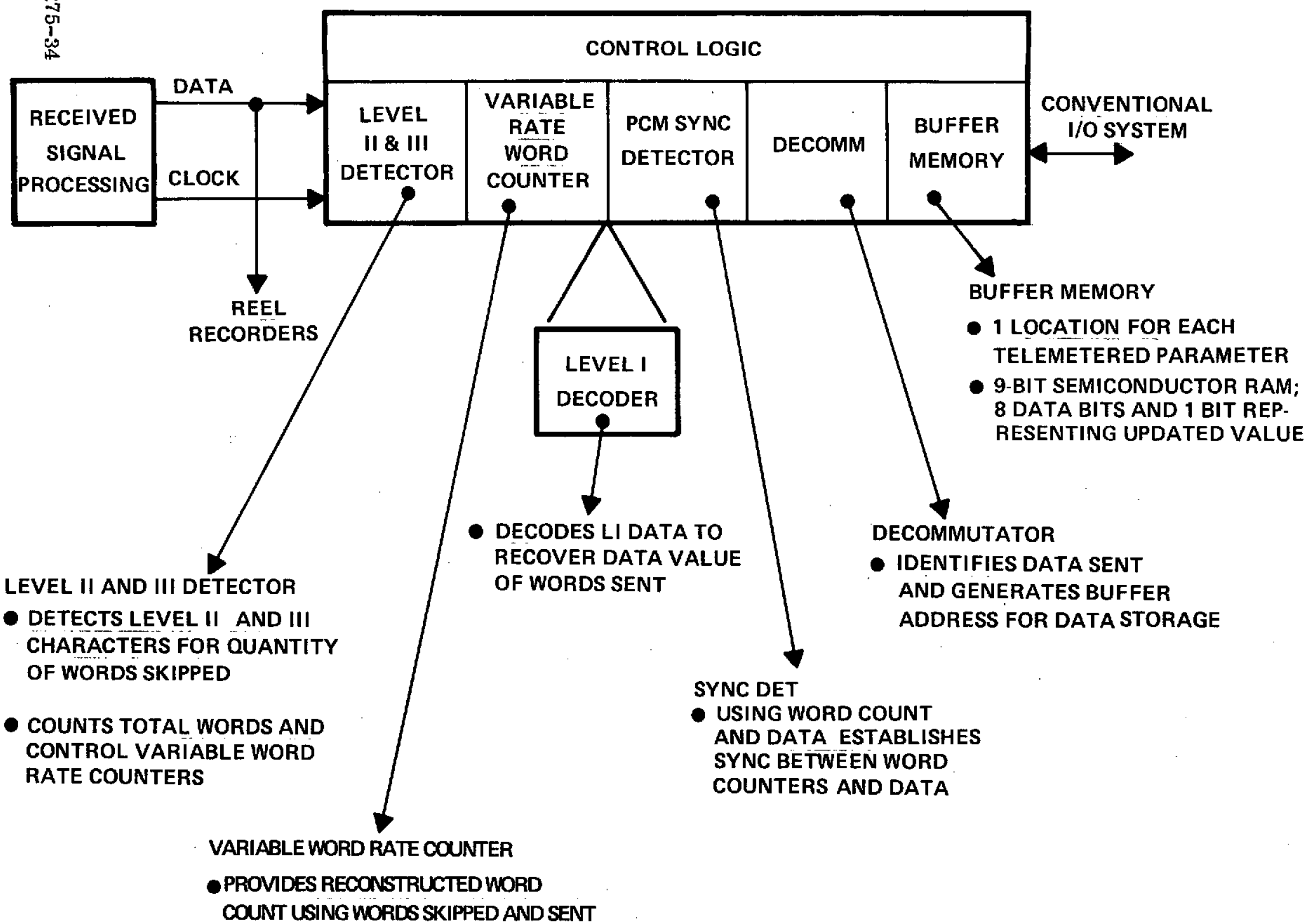


Figure 7. Ground Based Decompression System Functional Diagram

2.1.3 Conclusions

The telemetry portion of the Orbiter air/ground link can be compressed sufficiently to provide a significant increase in channel performance. The voice portion of the data link must be similarly compressed in order to achieve the desired performance increase. It is understood that NASA has separately contracted for a study concerned with voice compression.

2.1.4 Recommended Action

It is recommended that followon studies be initiated to permit an evaluation of the performance of Level I when combined with spread spectrum. It is anticipated that the longer transition intervals guaranteed by the Level I data will produce meaningful processing gains and enhance synchronization techniques.

2.2 TASK 3.2.2 REDUNDANCY REMOVAL FOR ORBITER GROUND-TO-GROUND LINK

2.2.1 Purpose and Scope

The purpose of this primary task is to determine the potential for using data compression techniques to provide improved ground link data throughput and to consider methods for improving the integrity of the transmitted data signal through the use of Delco three-phase encoding. Unlike the air-to-ground data link, these channels are bandwidth limited, and data compression is desirable to decrease the cost of multiple, wideband data links, which would be necessary to convey the uncompressed source data. In order to accomplish the intent of this task, the following items were accomplished:

- Bit error rate studies to compare Delco encoding with conventional methods
- Estimate the redundancy content of Orbiter PCM data
- Determine achievable throughput using data compression
- Evaluate the compatibility of Delco encoding techniques with channel encoding and decoding
- Generate a preliminary mechanization that illustrates the method of incorporation of selected techniques.

2.2.2 Significant Results Obtained

2.2.2.1 Bit Error Rate Studies

2.2.2.1.1 Level I in Bandwidth Limited Applications

Section 2 of Appendix B is devoted to the definition and analysis of an optimum Level I system for bandwidth limited applications. The optimum system identified is very similar to the Delco laboratory system. The two systems differ, at most, by nonessential differences in the transmitting and receiving filters. This is an important fact. The optimum Level I system transmits data at the Nyquist rate; that is, at a rate which is twice the bandwidth of the channel. The laboratory system also achieves transmission at the Nyquist rate utilizing simple, low cost hardware. Most systems operating over bandwidth-limited channels operate at only one-half to two-thirds the Nyquist rate in order to reduce complexity and simplify the hardware requirements. Delco's Level I achieves full Nyquist rate transmission without significant reduction in error performance.

Table 1 illustrated the basic results. Level I in bandwidth-restricted applications has only 1.8 dB less margin over noise than fully expanded Level I. The results of Table 2 may, therefore, be adapted to bandwidth-restricted Level I by adding 1.8 dB to each of the signal-to-noise ratio entries there.

2.2.2.1.2 Delco Electronics 3-Phase Versus Conventional 4-Phase Differential Phase Modulation

An analytical expression was developed for phase error probability as a function of signal-to-noise ratio for L-phase differential phase modulation (DPM) and evaluated numerically for biphase, 3-phase, and 4-phase modulation at signal-to-noise ratios ranging from 0 to 15 dB. Subsequently, the effectiveness of Delco Electronics 3-phase versus conventional 4-phase DPM was determined. (Refer to Appendix C.)

The results indicate that, for example, at a SNR of 10 dB, the probability of a phase error in 3-phase modulation is down by a factor of ten from that of 4-phase. As SNR increases, this factor increases, and the advantages of 3-phase modulation become more prominent. Translating phase error rate into bit error rate, our results for conventional 4-phase modulation show that a single phase error produces on the average 1 bit error; for Delco's 3-phase modulation, a phase error produces on the average approximately 1-1/2 bit errors. Comparing the two techniques at 10 dB, the 3-phase modulation bit-error rate is down by a factor of about seven from that of 4-phase modulation.

2.2.2.1.3 Compatibility of Convolutional and Level I Encoding

A study was conducted to determine if a convoluted code could be encoded in Level I without impairing the error correcting effectiveness of Viterbi decoding. Rate one-half convolutional encoding of random source data was used for the investigation.

Two digital computer programs were employed in error-correcting simulations. (Refer to Appendix D, Section 5.) The first permits rate one-half convolutional encoding, error injection, and Viterbi decoding. The second incorporates rate one-half convolutional encoding followed by Level I encoding, error injection, Level I decoding (using the Viterbi decoding algorithm), and Viterbi decoding.

Simulations were made with a source data sequence of 4,500 random bits. Error rates were established which were high enough to result in errors in the decoded sequence. For random source data, Level I encoding leads effectively to a bandwidth expansion by a factor of two. Taking this into account through the expression for the probability of a channel error in terms of bandwidth, the results showed that the error correcting ability of Viterbi decoding is not decreased at high error rates. Below an error rate of about 30%, the error correcting ability of Viterbi decoding is, however, degraded through the introduction of Level I.

2.2.2.2 Redundancy Estimate

2.2.2.2.1 Priority Transmission – Uncompressed Air/Ground Link

In the event that compression is not used for the downlink, priority transmission can provide for the real time transmission of vital data followed by nonreal time transmission of less critical and/or redundant information.

Critical data consists of out-of-limits signals, changes in the state of discrete signals, and samples of signals selected for consumables management and other monitoring tasks. The percentage of signals of real time interest is very small – it has been estimated that a maximum of 5% of the total monitored signals will display out-of-limits conditions for Orbiter malfunctions, and these will persist only until redundant Orbiter systems are placed on-line to correct the abnormal condition. This percentage will result in approximately 800 words per second which are candidates for transmission; further processing to remove the redundancy in these signals will reduce the candidates significantly such that they can easily be accommodated by a conventional 4.8 kb/s telephone transmission link. Appendix A contains the results of the redundancy reduction achieved for telemetered data signals, and it is reasonable to assume that out-of-limits channels will exhibit the same average change history as in-limits signals. Nonreal time transmission of the accumulated flight data is accomplished using a ±1, "one" bit change criteria, which according to the data in Appendix A, permits transmission of the entire data set well before the next orbital pass.

2.2.2.2.2 Priority Transmission – Compressed Air/Ground Link

When compression is used in the downlink system, the degree of redundancy eliminated will necessarily be less than that which can safely be used in the ground link. Data eliminated from the downlink is generally not recoverable for post-test usage, which is not true of the ground link system. All data received from the downlink can be recorded for later use, which permits greater latitude in the ground compression system. This permits using redundancy detection with wider apertures and heavier filtering than would be prudent for the downlink case. As in the uncompressed downlink case, limit detectors and discrete processors provide for real time transmission of critical data and nonreal time, which are compatible with standard telephone data transmission techniques.

2.2.3 Mechanization

The processing subsystem selected is one which has sufficient speed to assure evaluation of the full 16k words/second contained in the uncompressed downlink data. These processors are of the type produced by Delco for several military programs including the TDMG and SSTC equipments, and are identical to those proposed for the processor option for the Space Shuttle PCMMU. (Reference Delco proposal P74-2-6-1.) The functional mechanization for the ground-based compressor system is shown in Figure 8.

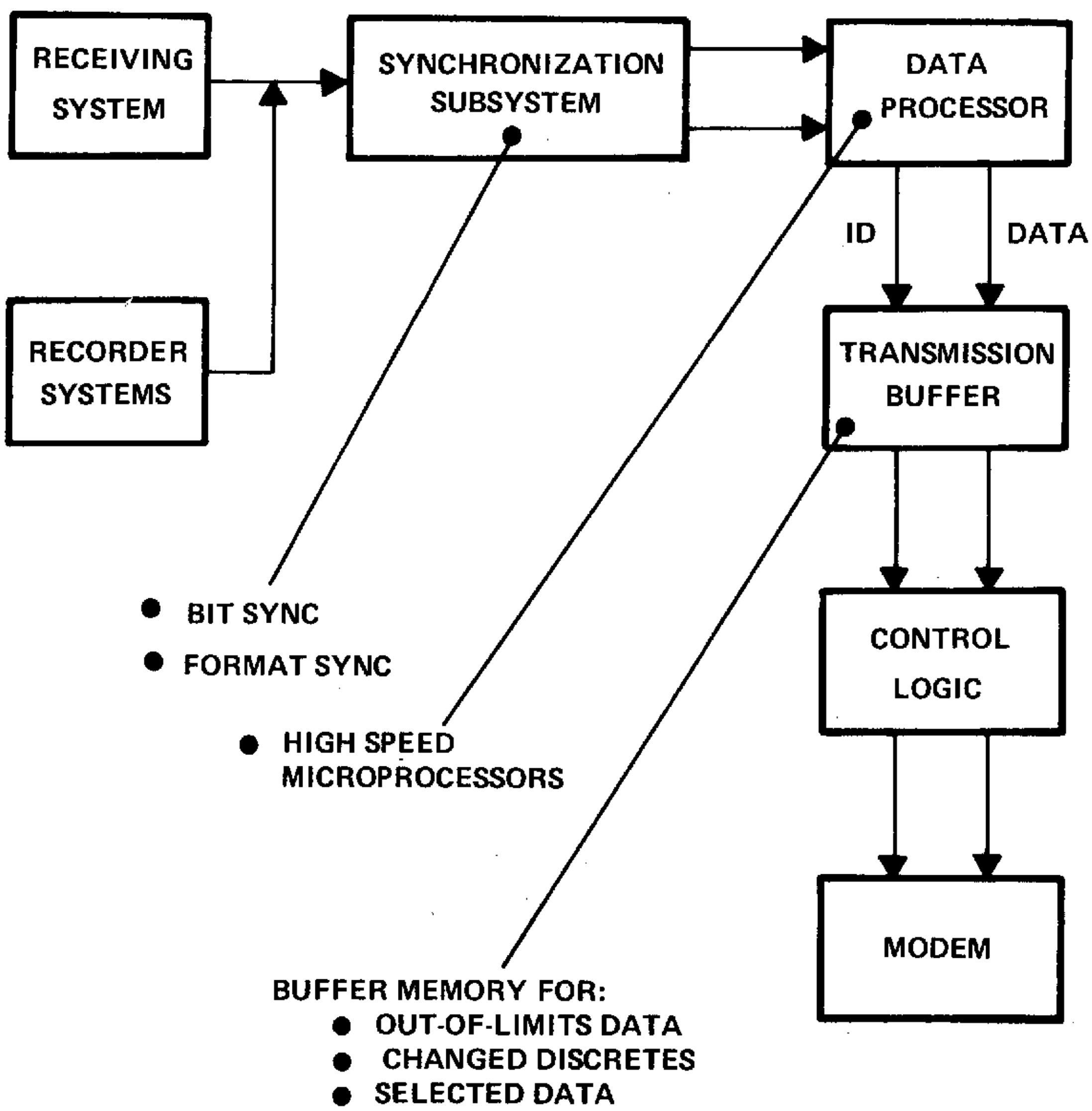


Figure 8. Ground Based Data Compressor Basic Mechanization

2.2.4 CONCLUSIONS

The ability to store real time data at the receiving terminals for delayed transmission significantly reduces the real time throughput requirement if sufficient signal processing is provided to extract signals of immediate interest. High speed telemetry data preprocessors exist for this purpose; those proposed for the PCMMU preprocessors are examples of proven techniques. Use of these preprocessors for selection of out-of-limits channels, changes in discrete signals, and other signals of interest provides for a very low real time throughput requirement. For this application, the throughput rate necessary to support the program can be accomplished without the use of Level I or Delco 3-phase DPSK. These encoding systems have been shown to possess very desirable characteristics for telephone data transmission; however, completely adequate transmission rates can be provided for compressed data using conventional 4-phase DPSK and commercially available data MODEMS.

2.2.5 RECOMMENDATIONS

In view of the vital role played by the telemetry data preprocessors in the ground data link, it is recommended that a study be undertaken to prepare a detailed specification for the ground-based telemetry preprocessor unit.

APPENDIX A

SAMPLE PCM FORMAT
USING SELECTED TITAN IIIC DATA SIGNALS

1.0 INTRODUCTION

The results of the analysis performed to determine the expected redundancy content of the Orbiter link are summarized herein. Actual Titan IIIC flight data was subjected to the data compression technique of determining the percent of data that exceeded programmable apertures. As a result of this analysis, additional methods and mechanization modifications were identified and investigated, the results of which are also included herein.

2.0 APPROACH

Actual Titan IIIC telemetry data, selected from the 30 May 1974 launch, was used for the redundancy assessment tests via a Sigma 7 computer simulation. The basis for selection of the 33 vehicle and guidance analog measurements, provided in Table 1, was to provide:

- Similarity to envisioned type of orbiter data
- A mixture of sampling rates
- A mixture of quiescent and highly active signals
- Signal signatures that are dependent upon environmental conditions.

The types of data utilized consisted of power supplies, discretes, steering/actuator functions, temperatures, pressures, accelerations, vehicle rate, and IGS bus voltages and currents. All measurement data types were subjected to the data compression techniques over 30 to 50 seconds of the following four desired environmental periods: (1) quiescent prelaunch (lg), (2) maximum dynamic pressure, (3) maximum thrust (approximately 4.5g), and (4) coast.

The Sigma 7 computer program simulated the redundancy reduction technique of only transmitting data values when a programmed allowable aperture was exceeded. Upon exceeding the allowable aperture, an updated tolerance, with the transmitted data value equal to the new nominal, was derived and used for subsequent transmission tests. Various reasonable

SIGNAL TYPE	SAMPLE RATE PER SECOND
Roll, Pitch, Yaw Rate Gyro Outputs	400
Pitch, Yaw Lateral Accelerometer Outputs	400
IGS Voltage and Current	20
SA 1, 2, and 3 Thrust Chamber Pressures	400
IL 1, 2, and 3 Current Ladder Outputs	100
EL 1 and 2 Voltage Ladder Outputs	100
BL 2 Sequence System Discretes	100
BL 6 and 7 ACS Nozzle Discretes	100
Guidance Truss (MGC) Temperature	100
Guidance Truss (IMU) Temperature	20
Four ACS Nozzle Pressures	200
X, Y, Z Payload Accelerometers	400
IMU Internal Temperature	20
IMU -15 Vdc Unregulated	20
IMU +15 Vdc Regulated	20
IMU 28 Vac, 0 Phase	20
MGC -6 Vdc	20
TPS Bus Current	800

Table 1. Signals Tested for Redundancy

magnitudes were evaluated to aid in determining the minimum bit rate required to support the transmission of orbiter data.

3.0 DISCUSSION

The results of subjecting the actual Titan IIIC flight data to the data compression technique of determining percent of data that exceeded programmable apertures is provided in Tables 2 through 7. The Sigma 7 computer output was grouped into the categories listed on each of the tables. The group average percent per flight period is summarized for each aperture tested.

Note: Tables 2 through 13 and Figures 1 and 2 appear at the end of this appendix.

The differences in signal signatures is evident in comparing the summarized group averages of active signals, provided in Tables 2 and 3, with those of very quiescent signals, provided in Tables 6 and 7. Hardcopy chart recordings of each of the signals were made to aid in the Sigma 7 program checkout, and an example of some of the active signals during the maximum dynamic pressure period is provided in Figure 1.

Examination of the results initially tabulated illustrated that several signals exceeded the ± 1 bit aperture for a significant percentage of the time, while the actual signal only exceeded ± 1 bit less than 2% of the time. This was due to the fact that the initially stored signal value was at a peak value rather than the true nominal value. This problem was alleviated by programming additional initialization criteria, where the nominal for the first aperture was derived from the average of the samples contained in the first record.

The importance of electrically terminating defunct signals when incorporating a data compression scheme such as that tested was clearly illustrated in several of the 33 signals; these signals are footnoted on the tables by an asterisk. These conditions occurred during the parking orbit coast phase with detached Stage II signals, including a vehicle roll rate gyro and a thrust chamber pressure. The variations exceeded the allowable ± 3 counts 99.8% of the time for one signal and 75% of the time for the other signal. Obviously, conditions like these would severely defeat data compression objectives.

The total percentage of all 33 signals exceeding the tested apertures per flight phase was as follows.

	± 1 Bit	± 2 Bits	± 3 Bits	± 5 Bits
Prelaunch	8.8	6.6	5.5	4.0
Maximum Q	24.1	19.2	16.3	12.1
Maximum Thrust	14.5	7.3	4.5	2.2
Coast*	16.1	10.3	10.2	10.0
Coast	5.7	0.3	0.2	0.1

* Includes percentages of noisy defunct signals.

As is apparent when reviewing the group average percentages per flight phase for each of the categorized six groups, primary emphasis should be placed on the most active signals, which are summarized in Tables 2 and 3. Thus, additional logic was defined and implemented in the Sigma 7 simulation to define exceeding a programmable aperture only when this occurred for a programmable number of consecutive samples. Tables 8 and 9 contain the results of the following three tests: (1) greater than 1 bit for two consecutive samples, (2) greater than 2 bits for two consecutive samples, and (3) greater than 1 bit for three consecutive samples. Direct comparison can be made between Tables 2 and 8 and between Tables 3 and 9. The average percentage of the four groups of active IGS data exceeding the ± 1 bit aperture (Table 2) was 48% compared with 22% of the same data exceeding the ± 1 bit for two consecutive samples (Table 8). Likewise, the reduction for the ± 2 bit is 27% compared with 9%. The percentage of data exceeding the greater than 1 bit for three consecutive samples is comparable to that of the greater than 2 bit for two consecutive samples. Similar reductions are apparent for the active airframe measurements, where 48% and 38% (Table 3) are reduced to 22% and 18% (Table 9) for the ±1 and ±2 bit cases, respectively.

The significant advantages of incorporating the consecutive sample logic is further supported by the fact that the noise outputs of the defunct signals during coast (footnoted by an asterisk) are suppressed. Although the ± 5 bit aperture without consecutive logic would provide a similar data compression ratio as the ± 2 bit with consecutive logic, important signal characteristics could be obscured if the signal sensitivities were optimally defined.

Mean and standard deviations of the time between aperture excursions were also computed for all 33 signals. However, the usefulness of the statistical outputs was very limited. Significance of the mean tests were only satisfied for the very active IGS and airframe measurements. These statistics for the active IGS measurements are provided in Table 10. When no aperture excursions were recognized, the mean is equal to the sample size, and the standard deviation is zero. Also, the mean has been rounded and the standard deviation truncated.

4.0 ADDITIONAL INVESTIGATIONS

The average string length of zeros, documented herein, was calculated to determine a compression ratio that would be realized from a data set equal to the 33 signals tested at the Titan IIIC telemetry sampling rates.

The format of the repeatable block of 353 samples

$$\left[\frac{1}{20}\sum_{i=1}^{33} (SPS_i)\right]$$

is provided in Table 11. The percentage of data transmitted was that resulting from exceeding the recommended greater than 2 bits for two consecutive samples data compression aperture.

A very large compression ratio would be realized during coast since the only data exceeding aperture was 23% and 6% of two 20 S/s signals, 1% of two 100 S/s signals, and 1% of one 200 S/s signal. With these very few aperture excursions, one sample of one signal would be transmitted only 49% of the time. Thus, for the worst case, 51% of the time the string length of zeros would be the total block of 360 words. With random distribution, the average string length during coast for the 360-word block would be approximately 340 words, which would yield a compression ratio of 100:1.

To determine the achieveable compression for more active periods, the percentage of data exceeding the subject aperture during the period of maximum dynamic pressure (max. Q) was obtained. An illustration of the transmitted data in the 360-word format is provided in Table 12. The calculations of the average string length during maximum Q are tabulated in Table 13, and the result is plotted in Figure 2. As illustrated, the average string length leveled out at 18.85 after a sample size greater than 80. This average string length would result in an average compression ratio of 10:1.

5.0 CONCLUSION

Dependent upon the degree of data compression desired, either the greater than 1 bit or greater than 2 bits for two consecutive samples should be implemented in the transmission of Orbiter data. The greater than 2 bits for two consecutive samples data compression technique is recommended to provide the most meaningful data at a highly desirable compression ratio. Ratioing these resulting percentages of the data compression technique selected by the number, type, and sample rate of measurements planned for Orbiter data will determine the predicted data compression and required bit rate for transmission of Orbiter data.

PHASE	SIGNAL RATE; #; NAME	NUMBER SAMPLES	+/- 1 BIT	+/- 2 BITS	+/- 3 BITS	+/- 5 BITS
			********** APERTURE / PERCENT OF TOTAL SAMPLES **********			
PRELAUNCH	20; 1180;	603	148 / 24.5	107 / 17.7	48 / 8.0	14 / 2.3
MAXIMUM Q	IGS Volts	1003	280 / 27.9	181 / 18.0	85 / 8.5	20 / 2.0
MAX. THRUST		484	125 / 25.8	53 / 11.0	8 / 1.7	8 / 1.7
COAST		440	100 / 22.7	54 / 12.3	23 / 5.2	13 / 2.9
PRELAUNCH	20; 1181;	603	447 / 74.1	421 / 69.8	372 / 61.7	191 / 31.7
MAXIMUM Q	IGS Current	1003	952 / 94.9	907 / 90.4	763 / 76.1	321 / 32.0
MAX. THRUST		484	443 / 91.5	364 / 75.2	274 / 56.6	215 / 44.4
COAST		440	267 / 60.7	191 / 43.4	138 / 31.4	120 / 27.0
PRELAUNCH	20; 322;	603	230 / 38.1	14 / 2.3	0 / .0	0 / .0
MAXIMUM Q	MGC-6Vdc	1003	501 / 50.0	2 / .2	0 / .0	0 / .0
MAX. THRUST		484	241 / 49.8	22 / 4.5	0 / .0	0 / .0
COAST		440	24 / 5.5	0 / .0	0 / .0	0 / .0
PRELAUNCH	20; 228;	603	157 / 26.0	115 / 19.1	0 / .0	0 / .0
MAXIMUM Q	Temp., IMU Plat.	1003	437 / 43.6	0 / .0	0 / .0	0 / .0
MAX. THRUST		484	173 / 35.7	62 / 12.8	0 / .0	0 / .0
COAST		440	299 / 68.0	204 / 46.4	132 / 30.0	0 / .0
Group Avg. % for:						
Prelaunch			40.7	27.2	17.4	8.4
Max. Q			54.1	27.2	21.2	8.5
Max. Thrust			50.7	25.9	14.6	11.5
Coast			39.2	25.5	9.2	7.6

Table 2. **Active IGS Measurements Throughout Flight**

ORIGINAL PAGE IS
OF POOR QUALITY

PHASE	SIGNAL RATE; #; NAME	NUMBER SAMPLES	********** APERTURE / PERCENT OF TOTAL SAMPLES +/- 1 BIT	+/- 2 BITS	+/- 3 BITS	********** +/- 5 BITS
PRELAUNCH	400;1248;	12060	2949 / 24.5	2001 / 16.6	1506 / 12.5	885 / 7.3
MAXIMUM Q	Stg. II	20060	13126 / 65.4	9764 / 48.7	7459 / 37.2	4332 / 21.6
MAX. THRUST	P Rate Gyro	9680	1263 / 13.0	343 / 3.5	141 / 1.5	8 / .1
COAST		8800	*3319 / 37.7	0 / .0	0 / .0	0 / .0
PRELAUNCH	400;1249;	12060	2722 / 22.6	1861 / 15.4	1346 / 11.2	741 / 6.1
MAXIMUM Q	Stg. II	20060	10775 / 53.7	7323 / 36.5	5192 / 25.9	2814 / 14.0
MAX. THRUST	Y Rate Gyro	9680	3540 / 36.6	2220 / 22.9	1501 / 15.5	791 / 8.2
COAST		8800	0 / .0	0 / .0	0 / .0	0 / .0
PRELAUNCH	400;1250;	12060	3381 / 28.0	2487 / 20.6	1980 / 16.4	1274 / 10.6
MAXIMUM Q	Stg. II	20060	14830 / 73.9	12028 / 60.0	9765 / 48.7	6410 / 32.0
MAX. THRUST	R Rate Gyro	9680	2557 / 26.4	1112 / 11.5	553 / 5.7	130 / 1.3
COAST		8800	*8779 / 99.8	8779 / 99.8	8779 / 99.8	8860 / 99.8
PRELAUNCH	400;1251;	12060	4347 / 36.0	3570 / 29.6	3256 / 27.0	2777 / 23.0
MAXIMUM Q	LASS	20060	18652 / 93.0	17694 / 88.2	16794 / 83.7	15085 / 75.2
MAX. THRUST	Pitch	9680	6289 / 65.0	4612 / 47.6	3400 / 35.1	1970 / 20.4
COAST		8800	0 / .0	0 / .0	0 / .0	0 / .0
PRELAUNCH	400;1252;	12060	3632 / 30.1	3179 / 26.4	2845 / 23.6	2304 / 19.1
MAXIMUM Q	LASS	20060	17791 / 88.7	16377 / 81.6	15141 / 75.5	12754 / 63.6
MAX. THRUST	Yaw	9680	4215 / 43.5	2443 / 25.2	1540 / 15.9	608 / 6.3
COAST		8800	0 / .0	0 / .0	0 / .0	0 / .0
Group Avg. % for:						
Prelaunch			28.2	21.7	18.1	13.2
Max. Q			74.9	63.0	54.2	41.3
Max. Thrust			36.9	22.1	14.7	7.3
*Coast			27.5	20.0	20.0	20.1

*Would be zero, with proper electrical termination of defunct signals.

Table 3. Powered Flight, Active Airframe Measurements

PHASE	SIGNAL RATE; #; NAME	NUMBER SAMPLES	********** APERTURE / PERCENT OF TOTAL SAMPLES ********** +/- 1 BIT	+/- 2 BITS	+/- 3 BITS	+/- 5 BITS
PRELAUNCH	100; 301	3015	66 / 2.2	51 / 1.7	40 / 1.3	35 / 1.2
MAXIMUM Q	IL 1	5015	209 / 4.2	103 / 2.1	53 / 1.1	23 / .5
MAX. THRUST		2420	2 / .1	0 / .0	0 / .0	0 / .0
COAST		2200	0 / .0	0 / .0	0 / .0	0 / .0
PRELAUNCH	100; 302;	3015	75 / 2.5	54 / 1.8	44 / 1.5	34 / 1.1
MAXIMUM Q	IL 2	5015	165 / 3.3	55 / 1.1	42 / .8	15 / .3
MAX. THRUST		2420	3 / .1	0 / .0	0 / .0	0 / .0
COAST		2200	0 / .0	0 / .0	0 / .0	0 / .0
PRELAUNCH	100; 303;	3015	62 / 2.1	51 / 1.7	45 / 1.5	36 / 1.2
MAXIMUM Q	IL 3	5015	136 / 2.7	67 / 1.3	31 / .6	14 / .3
MAX. THRUST		2420	4 / .2	0 / .0	0 / .0	0 / .0
COAST		2200	0 / .0	0 / .0	0 / .0	0 / .0
PRELAUNCH	100; 310;	3015	134 / 4.4	115 / 3.8	103 / 3.4	25 / .8
MAXIMUM Q	EL 1	5015	214 / 4.3	170 / 3.4	135 / 2.7	17 / .3
MAX. THRUST		2420	0 / .0	0 / .0	0 / .0	0 / .0
COAST		2200	0 / .0	0 / .0	0 / .0	0 / .0
PRELAUNCH	100;311;	3015	114 / 3.8	30 / 1.0	26 / .9	19 / .6
MAXIMUM Q	EL 2	5015	291 / 5.8	191 / 3.8	137 / 2.7	55 / 1.1
MAX. THRUST		2420	2 / .1	2 / .1	0 / .0	0 / .0
COAST		2200	0 / .0	0 / .0	0 / .0	0 / .0
Group Avg. % for:						
Prelaunch			3.0	2.0	1.7	1.0
Max. Q			4.1	2.3	1.6	0.5
Max. Thrust			0.1	0.0	0.0	0.0
Coast			0.0	0.0	0.0	0.0

Table 4. Powered Flight, Steering Ladder Outputs

PHASE	SIGNAL RATE; #; NAME	NUMBER SAMPLES	********** +/- 1 BIT		APERTURE / PERCENT OF TOTAL SAMPLES +/- 2 BITS		+/- 3 BITS		********** +/- 5 BITS	
PRELAUNCH	800;1179;	24120	4 /	.0	0 /	.0	0 /	.0	0 /	.0
MAXIMUM Q		40120	0 /	.0	0 /	.0	0 /	.0	0 /	.0
MAX. THRUST	TPS Bus Current	19360	58 /	.3	39 /	.2	31 /	.2	17 /	.1
COAST		17600	0 /	.0	0 /	.0	0 /	.0	0 /	.0
PRELAUNCH	400;2350;	12060	0 /	.0	0 /	.0	0 /	.0	7 /	.1
MAXIMUM Q		20060	360 /	1.8	5 /	.0	0 /	.0	0 /	.0
MAX. THRUST	P/L Accel.	9680	149 /	1.5	16 /	.2	7 /	.1	2 /	.0
COAST	Long.	8800	0 /	.0	0 /	.0	0 /	.0	0 /	.0
PRELAUNCH	400;2351;	12060	0 /	.0	0 /	.0	0 /	.0	4 /	.0
MAXIMUM Q		20060	1615 /	8.1	546 /	2.7	308 /	1.5	82 /	.4
MAX. THRUST	P/L Accel.	9680	366 /	3.8	3 /	.0	3 /	.0	0 /	.0
COAST	Lat.	8800	0 /	.0	0 /	.0	0 /	.0	0 /	.0
PRELAUNCH	400;2352;	12060	0 /	.0	0 /	.0	0 /	.0	171 /	1.4
MAXIMUM Q		20060	4388 /	21.9	2480 /	12.4	1605 /	8.0	861 /	4.3
MAX. THRUST	P/L Accel.	9680	476 /	4.9	46 /	.5	12 /	.1	0 /	.0
COAST	Vert.	8800	0 /	.0	0 /	.0	0 /	.0	0 /	.0
PRELAUNCH	400;3010;	12060	0 /	.0	0 /	.0	0 /	.0	0 /	.0
MAXIMUM Q		20060	305 /	1.5	0 /	.0	0 /	.0	0 /	.0
MAX. THRUST	T/C Press SA3	9680	0 /	.0	0 /	.0	0 /	.0	0 /	.0
COAST		8800	*7328 /	83.3	6739 /	76.6	6628 /	75.3	6424 /	72.3
PRELAUNCH	400;3015;	12060	0 /	.0	0 /	.0	0 /	.0	0 /	.0
MAXIMUM Q	T/C Press SA1	20060	1 /	.0	0 /	.0	0 /	.0	0 /	.0
MAX. THRUST		9680	2753 /	28.4	722 /	7.5	118 /	1.2	20 /	.2
COAST		8800	0 /	.0	0 /	.0	0 /	.0	0 /	.0
PRELAUNCH	400;3016;	12060	0 /	.0	0 /	.0	0 /	.0	0 /	.0
MAXIMUM Q	T/C Press SA2	20060	19 /	.1	0 /	.0	0 /	.0	0 /	.0
MAX. THRUST		9680	2039 /	21.1	442 /	4.6	63 /	.7	20 /	.2
COAST		8800	*4609 /	52.4	0 /	.0	0 /	.0	0 /	.0
Group Avg. % for:										
Prelaunch				0.0		0.0		0.0		0.2
Max. Q				4.2		1.9		1.2		0.6
Max. Thrust				7.5		1.7		0.3		0.1
*Coast				17.0		9.6		9.4		9.1

*Would be zero, with proper electrical termination of defunct signals.

Table 5. High Sample Rate Vehicle Measurements

PHASE	SIGNAL RATE; #; NAME	NUMBER SAMPLES	********** APERTURE / PERCENT OF TOTAL SAMPLES ********** +/- 1 BIT	+/- 2 BITS	+/- 3 BITS	+/- 5 BITS
PRELAUNCH	100; 351;	3015	0 / .0	0 / .0	0 / .0	0 / .0
MAXIMUM Q	COD 1-8	5015	0 / .0	0 / .0	0 / .0	0 / .0
MAX. THRUST	B/L2	2420	1 / .0	1 / .0	1 / .0	1 / .0
COAST		2200	0 / .0	0 / .0	0 / .0	0 / .0
PRELAUNCH	20; 202;	603	31 / 5.1	0 / .0	0 / .0	0 / .0
MAXIMUM Q	-15 Vdc Unreg.	1003	51 / 5.1	0 / .0	0 / .0	0 / .0
MAX. THRUST		484	67 / 13.8	0 / .0	0 / .0	0 / .0
COAST		440	14 / 3.2	0 / .0	0 / .0	0 / .0
PRELAUNCH	20; 203;	603	89 / 14.8	0 / .0	0 / .0	0 / .0
MAXIMUM Q	+15 Vdc Reg.	1003	0 / .0	0 / .0	0 / .0	0 / .0
MAX. THRUST		484	80 / 16.5	0 / .0	0 / .0	0 / .0
COAST		440	101 / 23.0	0 / .0	0 / .0	0 / .0
PRELAUNCH	20; 208;	603	29 / 4.8	0 / .0	0 / .0	0 / .0
MAXIMUM Q	28 Vac. /0	1003	90 / 9.0	0 / .0	0 / .0	0 / .0
MAX. THRUST		484	0 / .0	0 / .0	0 / .0	0 / .0
COAST		440	87 / 19.8	0 / .0	0 / .0	0 / .0
PRELAUNCH	100; 2114;	3015	0 / .0	0 / .0	0 / .0	0 / .0
MAXIMUM Q	Temp. Guid.	5015	1 / .0	1 / .0	0 / .0	0 / .0
MAX. THRUST	Truss MGC	2420	0 / .0	0 / .0	0 / .0	0 / .0
COAST		2200	0 / .0	0 / .0	0 / .0	0 / .0
PRELAUNCH	20; 2113;	603	0 / .0	0 / .0	0 / .0	0 / .0
MAXIMUM Q	Temp. Guid.	1003	0 / .0	0 / .0	0 / .0	0 / .0
MAX. THRUST	Truss IMU	484	0 / .0	0 / .0	0 / .0	0 / .0
COAST		440	0 / .0	0 / .0	0 / .0	0 / .0
Group Avg. % for:						
Prelaunch			1.8	0.0	0.0	0.0
Max. Q			1.0	0.0	0.0	0.0
Max. Thrust			2.2	0.0	0.0	0.0
Coast			3.3	0.0	0.0	0.0

Table 6. Quiescent IGS Signals Throughout Flight

PHASE	SIGNAL RATE; #; NAME	NUMBER SAMPLES	********** APERTURE / PERCENT OF TOTAL SAMPLES ********** +/- 1 BIT		+/- 2 BITS		+/- 3 BITS		+/- 5 BITS	
PRELAUNCH	200;1587;	6030	0 /	.0	0 /	.0	0 /	.0	0 /	.0
MAXIMUM Q	T/C Press	10030	6 /	.1	4 /	.0	3 /	.0	2 /	.0
MAX. THRUST	N10 YL	4840	0 /	.0	0 /	.0	0 /	.0	0 /	.0
COAST		4400	0 /	.0	0 /	.0	0 /	.0	0 /	.0
PRELAUNCH	200;1577;	6030	2 /	.0	2 /	.0	2 /	.0	2 /	.0
MAXIMUM Q	T/C Press	10030	4 /	.0	3 /	.0	2 /	.0	1 /	.0
MAX. THRUST	N1 PU	4840	0 /	.0	0 /	.0	0 /	.0	0 /	.0
COAST		4400	46 /	1.0	37 /	.8	31 /	.7	24 /	.5
PRELAUNCH	200;1579;	6030	0 /	.0	0 /	.0	0 /	.0	0 /	.0
MAXIMUM Q	T/C Press	10030	3 /	.0	2 /	.0	1 /	.0	1 /	.0
MAX. THRUST	N7 PD	4840	0 /	.0	0 /	.0	0 /	.0	0 /	.0
COAST		4400	0 /	.0	0 /	.0	0 /	.0	0 /	.0
PRELAUNCH	200;1585;	6030	0 /	.0	0 /	.0	0 /	.0	0 /	.0
MAXIMUM Q	T/C Press	10030	4 /	.0	3 /	.0	3 /	.0	1 /	.0
MAX. THRUST	N4 YR	4840	0 /	.0	0 /	.0	0 /	.0	0 /	.0
COAST		4400	0 /	.0	0 /	.0	0 /	.0	0 /	.0
PRELAUNCH	100;385;	3015	0 /	.0	0 /	.0	0 /	.0	0 /	.0
MAXIMUM Q	ACS Discretes	5015	0 /	.0	0 /	.0	0 /	.0	0 /	.0
MAX. THRUST	B/L 6	2420	0 /	.0	0 /	.0	0 /	.0	0 /	.0
COAST		2200	23 /	1.0	23 /	1.0	23 /	1.0	23 /	1.0
PRELAUNCH	100; 391;	3015	0 /	.0	0 /	.0	0 /	.0	0 /	.0
MAXIMUM Q	ACS Discretes	5015	0 /	.0	0 /	.0	0 /	.0	0 /	.0
MAX. THRUST	B/L 7	2420	0 /	.0	0 /	.0	0 /	.0	0 /	.0
COAST		2200	18 /	.8	18 /	.8	18 /	.8	18 /	.8
Group Avg. % for:										
Prelaunch				0.0		0.0		0.0		0.0
Max. Q				0.0		0.0		0.0		0.0
Max. Thrust				0.0		0.0		0.0		0.0
Coast				0.4		0.4		0.3		0.3

Table 7. Quiescent Signals only Active During Coast

ORIGINAL PAGE IS OF POOR QUALITY

SIGNAL NUMBER	PHASE	SAMPLE RATE	NUMBER SAMPLES	>1 BIT FOR TWO CONSEC. SAMPLES	>2 BITS FOR TWO CONSEC. SAMPLES	>1 BIT FOR THREE CONSEC. SAMPLES
1180	PRELAUNCH	20	603	90 / 14.9	57 / 9.5	71 / 11.8
	MAXIMUM Q		1003	169 / 16.8	95 / 9.5	123 / 12.3
	MAX. THRUST		484	83 / 17.1	33 / 6.8	49 / 10.1
	COAST		444	54 / 12.2	26 / 5.9	45 / 10.1
1181	PRELAUNCH	20	603	191 / 31.7	184 / 30.5	106 / 17.6
	MAXIMUM Q		1003	292 / 29.1	245 / 24.4	201 / 20.0
	MAX. THRUST		484	195 / 40.3	155 / 32.0	131 / 27.1
	COAST		444	125 / 28.2	103 / 23.2	88 / 19.8
322	PRELAUNCH	20	603	90 / 14.9	0 / .0	1 / .2
	MAXIMUM Q		1003	500 / 49.9	0 / .0	0 / .0
	MAX. THRUST		484	188 / 38.8	0 / .0	0 / .0
	COAST		444	0 / .0	0 / .0	0 / .0
228	PRELAUNCH	20	603	26 / 4.3	8 / 1.3	1 / .2
	MAXIMUM Q		1003	157 / 15.7	0 / .0	17 / 1.7
	MAX. THRUST		484	39 / 8.1	0 / .0	8 / 1.7
	COAST		444	68 / 15.3	0 / .0	35 / 7.9
Group Avg. % for:						
	Prelaunch			16.4	10.3	7.4
	Max. Q			27.9	8.5	8.5
	Max. Thrust			26.1	9.7	9.7
	Coast			13.9	7.3	9.4

Table 8. Active IGS Measurements, Consecutive Logic

SIGNAL NUMBER	PHASE	SAMPLE RATE	NUMBER SAMPLES	>1 BIT FOR TWO CONSEC. SAMPLES	>2 BITS FOR TWO CONSEC. SAMPLES	>1 BIT FOR THREE CONSEC. SAMPLES
1248	PRELAUNCH	400	12060	1500 / 12.4	1123 / 9.3	668 / 5.5
	MAXIMUM Q		20060	6907 / 34.4	5346 / 26.7	3406 / 17.0
	MAX. THRUST		9680	682 / 7.0	149 / 1.5	224 / 2.3
	COAST		8880	* 150 / 1.7	0 / .0	0 / .0
1249	PRELAUNCH	400	12060	1451 / 12.0	1040 / 8.6	903 / 7.5
	MAXIMUM Q		20060	5619 / 28.0	3977 / 19.8	3300 / 16.5
	MAX. THRUST		9680	2269 / 23.4	1538 / 15.9	1668 / 17.2
	COAST		8880	0 / .0	0 / .0	0 / .0
1250	PRELAUNCH	400	12060	1615 / 13.4	1340 / 11.1	873 / 7.2
	MAXIMUM Q		20060	7597 / 37.9	6363 / 31.7	4406 / 22.0
	MAX. THRUST		9680	1625 / 16.8	717 / 7.4	886 / 9.2
	COAST		8880	* 1 / .0	1 / .0	2 / .0
1251	PRELAUNCH	400	12060	2036 / 16.9	1809 / 15.0	1281 / 10.6
	MAXIMUM Q		20060	9190 / 45.3	8638 / 43.3	5992 / 29.9
	MAX. THRUST		9680	3222 / 33.3	2660 / 27.5	2132 / 22.0
	COAST		8880	0 / .0	0 / .0	0 / .0
1252	PRELAUNCH	400	12060	1839 / 15.2	1618 / 13.4	1136 / 9.4
	MAXIMUM Q		20060	8831 / 44.0	8134 / 40.5	5679 / 28.3
	MAX. THRUST		9680	2432 / 25.1	1661 / 17.2	1640 / 16.9
	COAST		8880	0 / .0	0 / .0	0 / .0

Group Avg. % for:			
Prelaunch	14.0	11.5	8.0
Max. Q	38.0	32.4	22.7
Max. Thrust	21.1	13.9	13.5
*Coast	0.3	0.0	0.0

*Would be zero, with proper electrical termination of defunct signals.

Table 9. Active Airframe Measurements, Consecutive Logic

SIGNAL NUMBER	PHASE	SAMPLE RATE	NUMBER SAMPLES	+/- 1 BIT	+/- 2 BITS	+/- 3 BITS	+/- 5 BITS
				********** MEAN / STANDARD DEVIATION **********			
1180	PRELAUNCH	20	603	4 / 3	5 / 5	12 / 15	42 / 63
	MAXIMUM Q		1003	3 / 2	5 / 4	11 / 13	42 / 123
	MAX. THRUST		484	3 / 2	[illegible] / 9	52 / 128	53 / 128
	COAST		444	4 / 3	7 / 8	16 / 20	31 / 43
1181	PRELAUNCH	20	603	1 / 0	1 / 0	1 / 1	3 / 3
	MAXIMUM Q		1003	1 / 0	1 / 0	1 / 0	3 / 2
	MAX. THRUST		484	1 / 0	1 / 0	1 / 1	2 / 1
	COAST		444	1 / 0	2 / 1	3 / 2	3 / 3
322	PRELAUNCH	20	603	2 / 4	40 / 59	603 / 0	603 / 0
	MAXIMUM Q		1003	2 / 0	184 / 258	1003 / 0	1003 / 0
	MAX. THRUST		484	2 / 0	21 / 38	484 / 0	484 / 0
	COAST		444	17 / 23	444 / 0	444 / 0	444 / 0
228	PRELAUNCH	20	603	3 / 7	5 / 11	603 / 0	603 / 0
	MAXIMUM Q		1003	2 / 1	1003 / 0	1003 / 0	1003 / 0
	MAX. THRUST		484	2 / 2	7 / 16	484 / 0	484 / 0
	COAST		444	1 / 1	3 / 1	3 / 4	444 / 0

Table 10. Active IGS Measurements, Statistics

	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20
0	A	B1	B2	B3	B4	B5	B6	B7	B8	A	B9	B10	B11	C1	C2	C3	C4	D1	A	B1
20	B2	—	—	—	—	—	B8	A	B9	—	B11	D2	—	—	—	D6	A	B1	—	B3
40	B4	B4	—	—	B8	A	B9	—	B11	C1	—	—	C4	E1	A	B1	—	—	—	B5
60	B6	—	B8	A	B9	—	B11	D7	D8	D9	E2	E3	A	B1	—	—	—	—	—	B7
80	B8	A	B9	—	B11	C1	—	—	C4	D1	A	B1	—	—	—	—	—	—	B8	A
100	B9	—	B11	D2	—	—	—	D6	A	B1	—	—	—	—	—	—	B8	A	B9	B10
120	B11	C1	—	—	C4	E4	A	B1	—	—	—	—	—	—	B8	A	B9	—	B11	E5
140	D7	—	D9	E6	A	B1	—	—	—	—	—	—	B8	A	B9	—	B11	C1	—	C3
160	C4	D1	A	B1	—	—	—	—	—	—	B8	A	B9	—	B11	D2	—	—	—	D6
180	A	B1	—	—	—	—	—	—	B8	A	B9	—	B11	C1	—	—	C4	E7	A	B1
200	B2	—	—	—	—	—	B8	A	B9	—	B11	D7	—	D9	E8		A	B1	—	B3
220	B4	—	—	—	B8	A	B9	—	B11	C1	—	—	C4	D1	A	B1	—	—	—	B5
240	B6	—	B8	A	B9	—	B11	D2	—	—	—	D6	A	B1	—	—	—	—	—	B7
260	B8	A	B9	—	B11	C1	—	—	C4		A	B1	—	—	—	—	—	—	B8	A
280	B9	—	B11	D7	—	D9			A	B1	—	—	—	—	—	—	B8	A	B9	B10
300	B11	C1	—	—	C4	D1	A	B1	—	—	—	—	—	—	B8	A	B9	—	B11	D2
320	D3	—	—	D6	A	B1	—	—	—	—	—	—	B8	A	B9	—	B11	C1	—	C3
340	C4		A	B1	—	—	—	—	—	—	B8	A	B9	—	B11	D7	—	D9		
360																				

Where

A	=	1 - 800 samples per second
B_1 - B11	=	11 - 400 samples per second
C1 - C4	=	4 - 200 samples per second
D1 - D9	=	9 - 100 samples per second
E1 - E8	=	8 - 30 samples per second
		Spares

Table 11. Format of Signals Sampled

	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20
0	0	0	—	—	0	3	12	20	27	0	32	41	43	0	—	—	—	—	—	0
20	0	0	0	3	12	20	27	0	32	41	43	0	0	0	1	1	0	0	0	0
40	0	3	12	20	27	0	32	41	43	0	—	—	—	—	—	—	—	—	0	3
60	12	20	27	0	32	41	43	2	3	4	0	—	—	—	—	—	0	3	12	20
80	27	0	32	41	43	0	—	—	—	—	—	—	—	—	0	3	12	30	27	0
100	32	41	43	0	—	0	1	1	0	—	—	—	0	3	12	20	27	0	32	41
120	43	0	—	—	—	—	—	—	—	—	0	3	12	20	27	0	32	41	43	0
140	2	3	4	0	—	—	—	—	0	3	12	20	27	0	32	41	43	0	—	0
160	0	—	—	—	—	—	0	3	12	20	27	0	32	41	43	0	—	0	1	1
180	0	—	—	—	0	3	12	20	27	0	32	41	43	0	—	—	0	10	0	0
200	0	—	0	3	12	20	27	0	32	41	43	2	3	4	24		0		—	0
220	0	3	12	20	27	0	32	41	43	0	—	—	—	—	—	—	—	—	0	3
240	12	20	27	0	32	41	43	0	—	0	1	1	0	—	—	—	0	3	12	20
260	27	0	32	41	43	0	—	—	0		0	—	—	—	0	3	12	20	27	0
280	32	41	43	2	3	4			0	—	—	—	0	3	12	20	27	0	32	41
300	43	0	—	—	—	—	—	—	—	—	0	3	12	20	27	0	32	41	43	0
320	0	0	1	1	0	—	—	—	0	3	12	20	27	0	32	41	43	0	—	0
340	0		0	—	—	—	0	3	12	20	27	0	32	41	43	2	3	4		
360																				

Where

A	=	0
B1 - B11	=	0, 0, 0, 0, 3, 12, 20, 27, 32, 41, 43
C1 - C4	=	0
D1 - D9	=	0, 0, 0, 0, 1, 1, 2, 3, 4
E1 - E8	=	0, 0, 0, 0, 0, 0, 10, 24
		Spare

Table 12. Percent of Transmitted Data

SAMPLE SIZE (n)	PERCENT SAMPLE NOT TRANSMITTED	PROBABILITY OF STRING = n	AVERAGE STRING OF ZEROS
5	100	100.0	5.00
6	97	97.0	5.97
7	88	85.0	6.85
8	80	68.0	7.63
10	73	50.0	8.82
11	68	34.0	9.56
12	59	20.0	10.05
23	57	11.0	11.47
24	97	11.0	12.85
25	88	10.0	14.06
26	80	7.8	15.00
28	73	5.7	15.74
29	68	3.9	16.25
30	59	2.3	16.57
34	57	1.3	16.80
35	99	1.3	17.03
41	99	1.3	17.34
42	97	1.2	17.64
43	88	1.1	17.92
44	80	0.9	18.15
46	73	0.6	18.32
47	68	0.4	18.44
48	59	0.3	18.53
59	57	0.1	18.57
60	97	0.1	18.61
61	88	0.1	18.65
62	80	0.1	18.69
64	73	0.1	18.74
65	68	0.05	18.77
66	59	0.03	18.78
67	57	0.02	18.79
68	98	0.02	18.80
69	97	0.02	18.81
77	96	0.02	18.82
78	97	0.01	18.83
79	88	0.01	18.83
80	80	0.01	18.84
82	73	0.01	18.84
83	68	0.005	18.85

Table 13. String Length Calculations

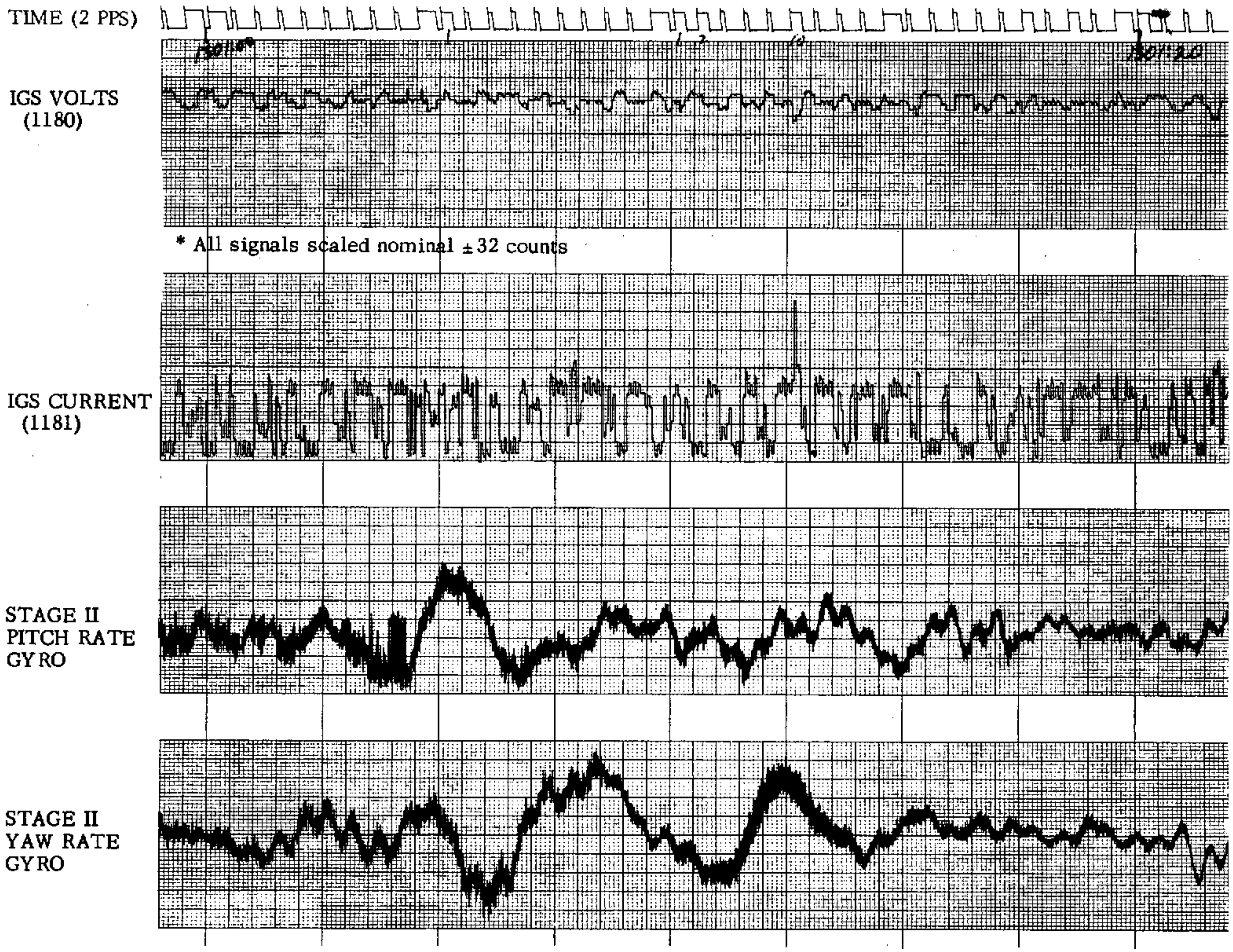


Figure 1. Signal Characteristics During Maximum Q

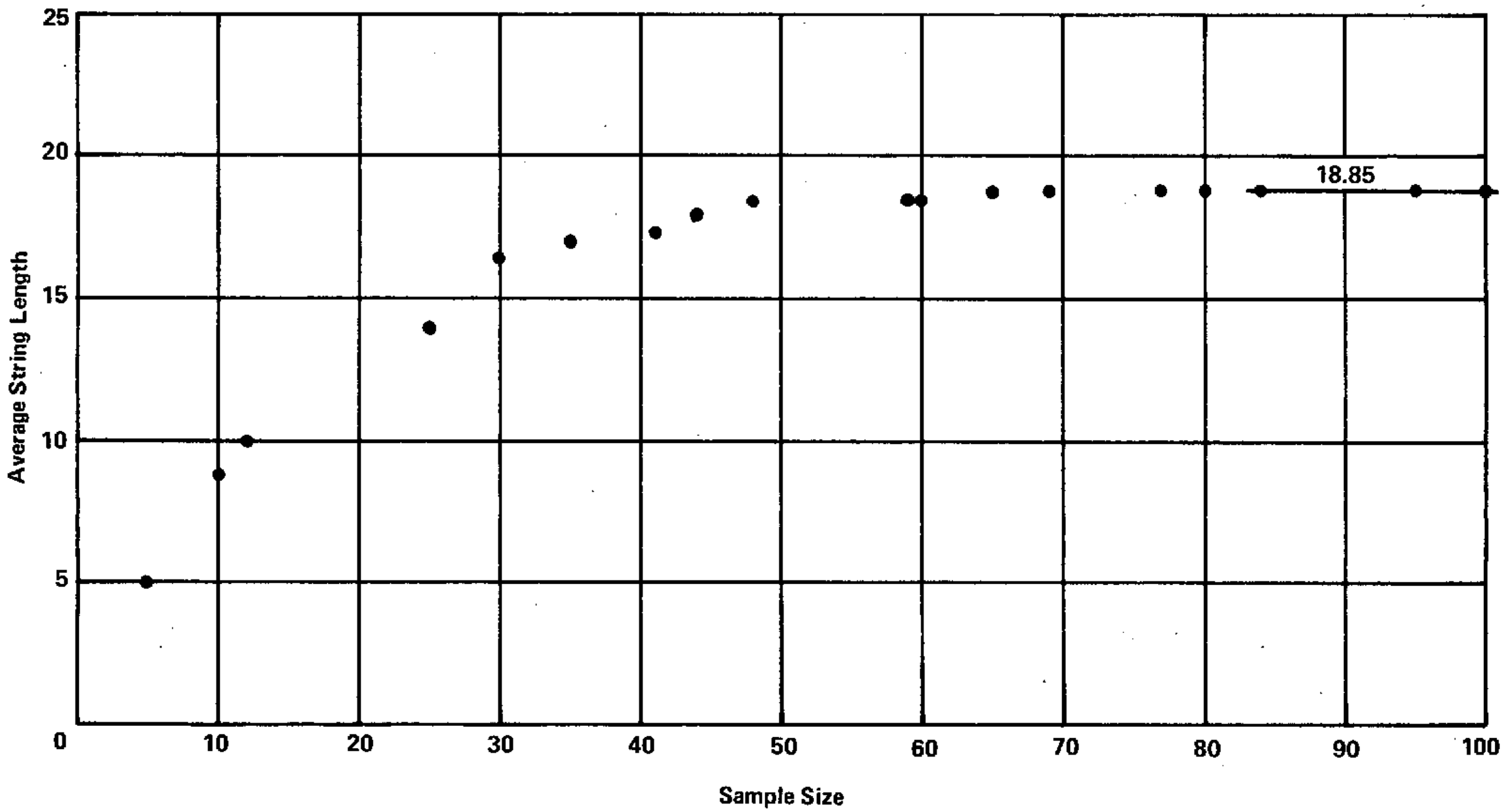


Figure 2. Average String During Maximum Q

APPENDIX B
LEVEL I THEORY

1.0 DESCRIPTION OF LEVEL I

1.1 Level I Algorithm

Level I encoding of binary digital data is applicable to amplitude-, frequency-, and phase-modulation systems as well as baseband systems. This appendix discusses the analysis of baseband systems employing Level I encoding. The methods utilized and the conclusions drawn may be expected to be applicable with some modifications to modulation systems.[1]

In the baseband case, the information is encoded by causing transitions between two voltage levels at times determined by the encoding algorithm. To facilitate the discussion, consider the Level I encoder to put out an analog voltage v(t) that is either +V or -V. (See Figure 1.)

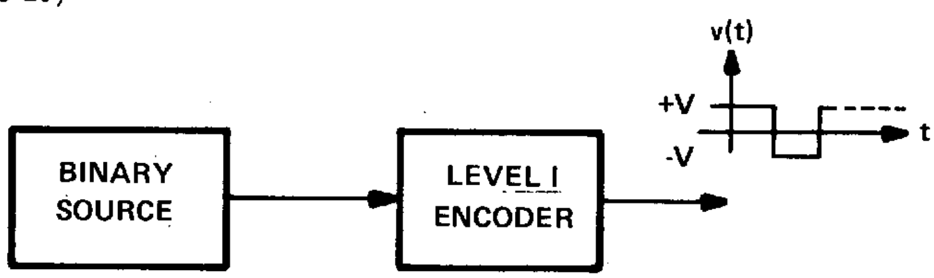


Figure 1. Level I Encoding

The binary source produces a logic 1 or 0 every T seconds. (The information rate is 1/T bits per second.) Now consider the time domain to be divided into cells (called bit cells) of length T seconds. The binary source associates a 1 or a 0 with each bit cell. The Level I encoder causes a transition between voltage levels ±V to occur at the beginning or at the middle of each bit cell, or no transition to occur, according to the following algorithm.

[1] Bennett, W.R. & Davey, J.R., Data Transmission, McGraw-Hill, N.Y. 1965, Chpt. 7-10 and particularly Chpt. 11.

1. To encode the first bit cell (starting in either level with x representing either a 0 or a 1):
   - (i) If the first three source bits are 11x or 000, encode a transition at the beginning of the bit cell.
   - (ii) If the first three source bits are 10x, encode a transition at the middle of the bit cell.
   - (iii) If the first three source bits are 01x or 001, encode no transition.

2. To encode the k^{th} bit cell ($k \geq 2$):
   - a. If the $(k-1)^{st}$ bit cell had no transition, apply the rules 1(i) through 1(iii) to source bits k, k+1, k+2.
   - b. If the $(k-1)^{st}$ bit cell had a transition at the middle of the cell, encode no transition.
   - c. If the $(k-1)^{st}$ bit cell had a transition at the beginning of the cell,
     - (i) If source bits k, k+1, k+2 are 1xx, encode no transition.
     - (ii) If source bits k, k+1, k+2 are 00x, encode a transition at the middle of the bit cell.

With one exception, these rules accomplish the encoding of transitions according to the information contained in pairs of source bits (with the appropriate constraints so that the information is encoded one-to-one). The exception is that three successive 0's are made to cause a transition at the beginning of the bit cell and another transition at the middle of the next bit cell. This exceptional encoding is provided to assure transitions in the encoded bit stream when a long string of 0's is present. With this one exception, the rules cause a transition at the beginning of the bit cell if the source pair is 11, a transition at the middle of the bit cell is the source pair is 10, and no transition if the source pair is 01 or 00. Except for the three 0's case, the constraint prevents transitions from occurring in successive bit cells. An example of a Level I encoded bit stream is given in Figure 2.

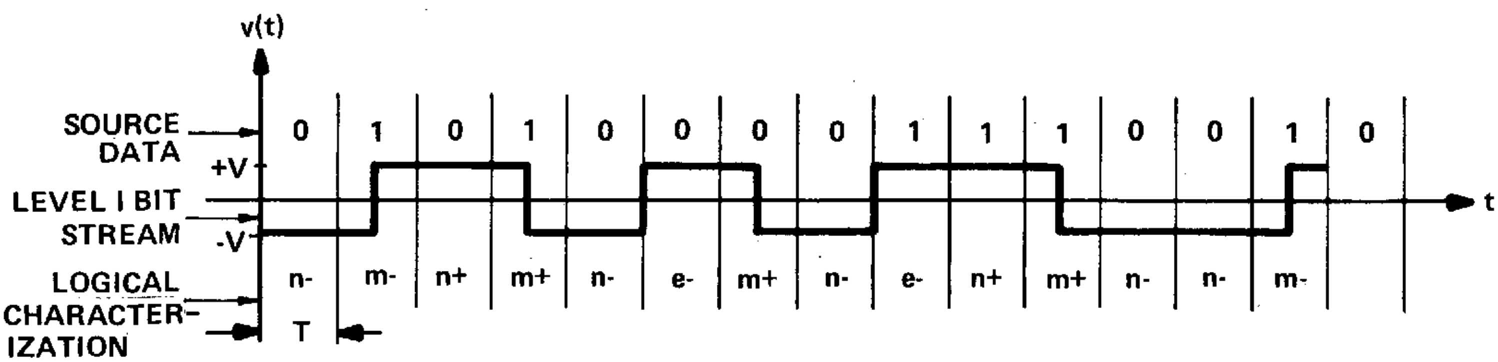


Figure 2. Level I Encoding, Example

1.2 Level I Probabilities

The Level I encoded bit stream will be analyzed as a stochastic process. The joint probabilities characterizing the process will be computed from the statistics of the binary source data and the Level I encoding algorithm. For this purpose, each encoded bit cell may be logically characterized as having an edge transition (e), a mid transition (m), or no transition (n) (as described in the algorithm), and, as having been in the upper level (+) or the lower level (-) at the beginning of the bit cell. (See Figure 2). (The analog signal v(t) making square transitions between voltage levels ±V, as in Figure 2, was introduced as a convenience in visualizing Level I and is not required to identify its statistical properties.)

The notation $p(\alpha\beta, k)$ with α = e, m, or n, and $\beta = \pm$, will be used for the probability that the k^{th} encoded bit cell is in the state $\alpha\beta$. For example, p(e+, 17) is the probability that the 17^{th} bit cell has had encoded an edge transition beginning from the upper level. Similarily, the notation $p(\alpha\beta, k; \alpha'\beta', k+1; \ldots\ldots; \alpha^{(s)}\beta^{(s)}, k+s)$ is used for the joint probabilities, and $p(\alpha^{(n+1)}\beta^{(n+1)}, k+n+1; \ldots\ldots; \alpha^{(s)}\beta^{(s)}, k+s \mid \alpha\beta, k; \alpha'\beta', k+1; \ldots\ldots; \alpha^{(n)}\beta^{(n)}, k+n)$ for the conditional probabilities. The latter is the probability that the $k+n+1^{st} \ldots, k+s^{th}$ bit cells are in the states $\alpha^{(n+1)}\beta^{(n+1)}, \ldots\ldots, \alpha^{(n+s)}\beta^{(n+s)}$ given that the $k^{th}, \ldots, k+n^{th}$ bit cells are known to be in the states $\alpha\beta, \ldots, \alpha^{(n)}\beta^{(n)}$, respectively.

Throughout this analysis, it is assumed that the 0's and 1's of the binary source occur with equal probability and that the information bits are statistically independent. In the case of statistically independent source data, the Level I algorithm makes it clear that the Level I bit stream is a Markov process because the content of the k^{th} encoded bit cell depends only on the content of the $(k-1)^{st}$. The conditional probability introduced above then satisfies the equation.

$$p(\alpha^{(n+1)}\beta^{(n+1)}, k+n+1; \ldots; \alpha^{(s)}\beta^{(s)}, k+s \mid \alpha\beta, k; \alpha'\beta', k+1; \ldots\ldots; \alpha^{(n)}\beta^{(n)}, k+n)$$
$$= p(\alpha^{(n+1)}\beta^{(n+1)}, k+n+1; \ldots\ldots; \alpha^{(s)}\beta^{(s)}, k+s \mid \alpha^{(n)}\beta^{(n)}, k+n) \quad (1)$$

Finally, it is assumed that the starting time has receded to minus infinity so that the Level I bit stream may be analyzed as a stationary process. In that case $p(\alpha\beta)$ may be written for $p(\alpha\beta, k)$, $p(\alpha\beta, \alpha'\beta', \ldots, \alpha^{(s)}\beta^{(s)})$ for

$$p(\alpha\beta k; \ldots\ldots; \alpha^{(s)}\beta^{(s)}, k+s), \text{ and } p(\alpha^{(n+1)}\beta^{(n+1)}, \ldots, \alpha^{(s)}\beta^{(s)} \mid \alpha^{(n)}\beta^{(n)})$$

for $p(\alpha^{(n+1)}\beta^{(n+1)}, k+n+1; \ldots\ldots; \alpha^{(s)}\beta^{(s)}, k+s \mid \alpha^{(n)}\beta^{(n)}, k+n)$.

Stationary Markov processes are completely characterized by $p(\alpha\beta)$ and the conditional probability $p(\alpha'\beta' \mid \alpha\beta)$, which is the probability that the bit cell in question is in the state $\alpha'\beta'$ given that the previous bit cell was known to be in the state $\alpha\beta$. This fundamental conditional probability $p(\alpha'\beta' \mid \alpha\beta)$ may be thought of as the transition probability from one bit cell to the next. To emphasize this interpretation, it is given the special notation

$$p(\alpha'\beta' \mid \alpha\beta) \equiv q(\alpha\beta, \alpha'\beta') \tag{2}$$

All the joint and conditional probabilities for the stationary Markov process may be written in terms of $p(\alpha\beta)$ and $q(\alpha\beta, \alpha'\beta')$. As an example, it may easily be shown that the s^{th} order joint probability may be written

$$p(\alpha\beta, \alpha'\beta', \ldots, \alpha^{(s)}\beta^{(s)}) = p(\alpha\beta)\, q(\alpha\beta, \alpha'\beta')\, q(\alpha'\beta', \alpha''\beta'') \ldots\ldots q(\alpha^{(s-1)}\beta^{(s-1)}, \alpha^{(s)}\beta^{(s)}) \tag{3}$$

The values of the transition probability matrix elements may be determined by inspection using the Level I algorithm and the statistical independence and equi-probability of the source 1''s and 0''s. They are

$$q(\alpha\beta, \alpha'\beta') = \begin{array}{cc} & \begin{array}{cccccc} e+ & m+ & n+ & e- & m- & n- \end{array} \leftarrow \alpha'\beta' \\ \begin{array}{c} e+ \\ m+ \\ n+ \\ \\ e- \\ m- \\ n- \end{array} & \left[\begin{array}{ccc|ccc} 0 & 0 & 0 & 0 & 1/3 & 2/3 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 3/8 & 1/4 & 3/8 & 0 & 0 & 0 \\ \hline 0 & 1/3 & 2/3 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3/8 & 1/4 & 3/8 \end{array}\right] \\ \uparrow & \\ \alpha\beta & \end{array} \tag{4}$$

To find the values of $p(\alpha\beta)$, it may be observed that p and q satisfy the equation

$$\sum_{\alpha\beta} p(\alpha\beta)\, q\,(\alpha\beta,\ \alpha'\beta') = p(\alpha'\beta') \tag{5}$$

This equation may be obtained as a limiting form of the Smoluchowski equation.[2]

The values of the elements of p obtained on solving Eq. (5) are

$$p\,(\alpha\beta) = \begin{array}{c} e+ \\ m+ \\ n+ \\ e- \\ m- \\ n- \end{array} \begin{bmatrix} 3/28 \\ 3/28 \\ 2/7 \\ 3/28 \\ 3/28 \\ 2/7 \end{bmatrix} \tag{6}$$

1.3 The Probability of Sampling Next to a Transition

In the system studied in Section 2 of this Appendix, the received Level I encoded wave train is first passed through a filter and then sampled in the middle of each half bit cell to determine which level the signal is in. The filter of Section 2 causes considerable rounding of the transitions (see Figure 2 where square transitions are shown) so that the sample values (in the absence of noise) are $\pm 1/2$ V rather than $\pm$ V if the sample is taken adjacent to a transition. It is, therefore, of interest to compute the probability that a given sample is taken next to a transition.

Each half of the bit cell must be considered separately. For samples made in the middle of the first half of a bit cell, the information is contained in $p(\alpha\beta)$. The sample is adjacent

2. M.C. Wang and G.E. Uhlenbeck, Rev. Mod. Phys. 17, 323 (1945). Reprinted in Wax, "Selected Papers on Noise and Stochastic Processes," Dover, 1954. - Take the limit $s \to \infty$ in Eq. (13) and use the discrete form of Eq. (4). (Note that Wang and Uhlenbeck use a sinistral convention in writing conditional probabilities, whereas I use a dextral convention except for q.)

to a transition if there was an edge or a mid transition; otherwise it is not. Therefore

$$\underset{1^{st}\text{half}}{\Pr}(\text{adjacent}) = p(e+) + p(m+) + p(e-) + p(m-) = 3/7 \tag{7}$$

For samples made in the middle of the second half of a bit cell, $p(\alpha\beta,\ \alpha'\beta')$ is required. The sample is adjacent a transition if a mid transition occurred in the first bit cell (assuming the sample was taken in the first bit cell) or if an edge transition occurred in the second bit cell. The probability is, therefore

$$\underset{2^{nd}\text{ half}}{\Pr}(\text{adjacent}) = p(m+,\ n-) + p(m-,\ n+) + p(n+,\ e+) + p(n-,\ e-) \tag{8}$$

Using Eqs. (3), (4), and (6), the values of $p(\alpha\beta,\ \alpha'\beta')$ are found to be

$$p(\alpha\beta,\alpha'\beta') = \begin{array}{c c} & \begin{array}{cccccc} e+ & m+ & n+ & e- & m- & n- \end{array} \\ \begin{array}{c} e+ \\ m+ \\ n+ \\ e- \\ m- \\ n- \end{array} & \left[\begin{array}{ccc|ccc} 0 & 0 & 0 & 0 & 1/28 & 1/14 \\ 0 & 0 & 0 & 0 & 0 & 3/28 \\ 3/28 & 1/14 & 3/28 & 0 & 0 & 0 \\ \hline 0 & 1/28 & 1/14 & 0 & 0 & 0 \\ 0 & 0 & 3/28 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3/28 & 1/14 & 3/28 \end{array}\right] \end{array} \tag{9}$$

Thus

$$\underset{2^{nd}\text{ half}}{\Pr}(\text{adjacent}) = 3/7 \tag{10}$$

Therefore, the result that will be used in Section 2 is that the sampler will measure $\pm 1/2\,V$ (in the absence of noise) 3/7 of the time and $\pm V$ 4/7 of the time.

2.0 ANALYSIS OF A LEVEL I SYSTEM

2.1 Description of the Delco Laboratory System

The system analyzed in this section corresponds closely to the system operating in the Delco laboratory. The system is shown in block diagram form in Figure 3.

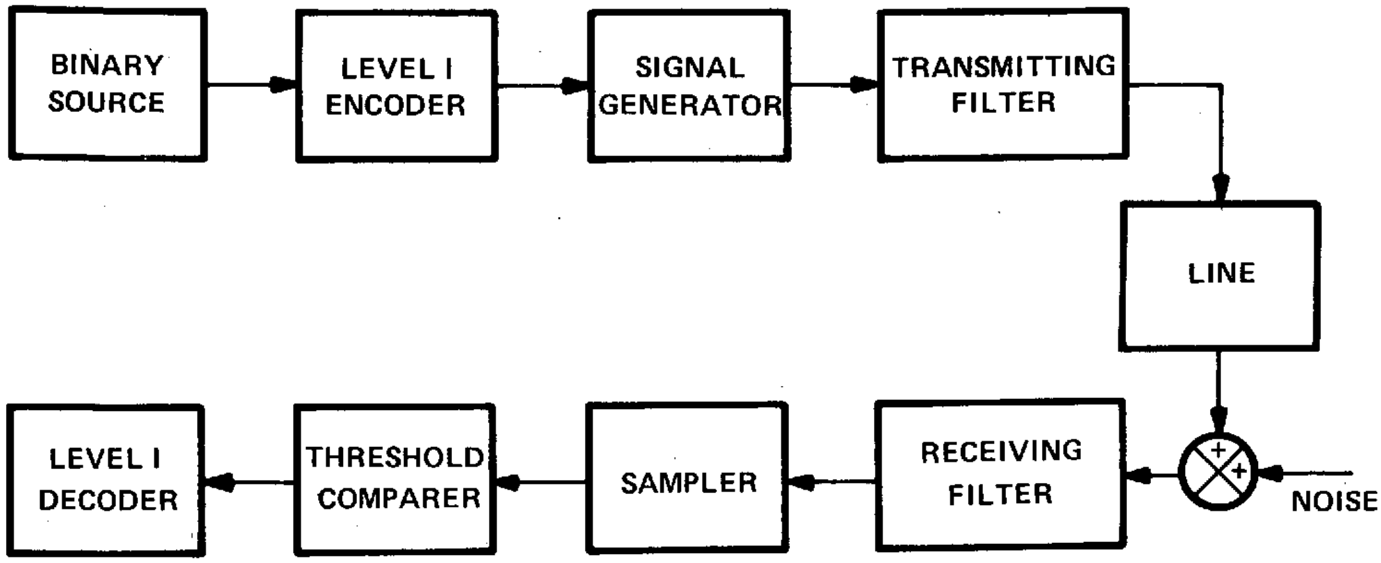


Figure 3. Delco Laboratory System – Idealized

The main idealization in the block diagram is that the actual system contains additional elements for clock alignment and baseline stabilization between the receiving filter and the sampler. Their effect on performance is neglected in this analysis, and perfect clock alignment and baseline stabilization are assumed.

The first three blocks of Figure 3 correspond to the two blocks of Figure 1. In Figure 3, the Level I encoder puts out a logical characterization, while the signal generator puts out the actual analog wave train such as is shown in Figure 2. The transmitting filter, line, and receiving filter cause a rounding of the square transitions shown in Figure 2, and it is the function of the sampler and threshold comparer to square the wave up so that it may be Level I decoded. The samples are taken in the center of each half-bit cell and compared against a zero threshold.

When the line imposes a bandwidth restriction and it is desired to transmit information at the highest rate consistent with that restriction (which is the usual application of Level I), the information rate, 1/T, is chosen to be the Nyquist rate; namely, twice the bandwidth of the line. The receiving filter is then chosen to have the same bandwidth (1/2T) so as to maximize the output signal-to-noise ratio. The transmission filter is usually omitted.

2.2 The System to be Analyzed — Notation

To define the system to be analyzed, the signals generated together with the transfer functions of the various filters of Figure 3 are specified. The signals and transfer functions are not (necessarily) chosen to correspond to those actually used in the laboratory, but rather are chosen to optimize the output signal-to-noise ratio at fixed transmitted power and to minimize intersymbol interference. This approach allows more general conclusions to be drawn from the results than would otherwise be the case. The method followed is that of Sunde (which generalizes earlier work of Nyquist) as described in Bennett and Davey (Footnote 1, Chapter 7).

For this analysis, the simplified block diagram of Figure 4 provides an adequate representation of the system.

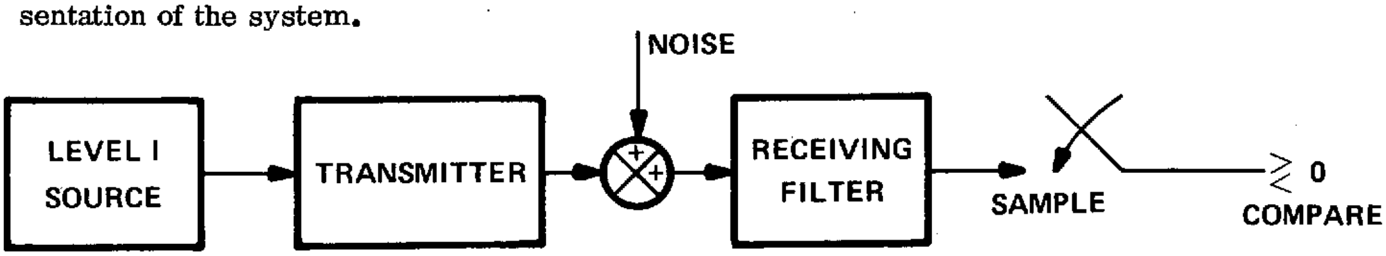


Figure 4. Simplified Block Diagram

The block marked "Level I Source" combines the binary source and the Level I encoder of Figure 3. Its output is a sequence of logical symbols characterizing the Level I coded information stream. For the present purposes, it is more convenient to use a different logical characterization than that shown in Figure 2 (that is, the characterization by α = e, m, or n; β = + or −). Each half-bit cell (of length T/2) is, therefore, assigned a +1 or a −1 depending on whether the voltage of Figure 2 is in the upper or the lower level during that half-bit cell. The +1 or −1 of the k^{th} half-bit cell is given the notation s_k. Then the complete Level I encoded bit stream may be logically characterized by the sequence $\{s_k\}$, $k = \ldots\ldots, -1, 0, 1, 2, \ldots\ldots$ The joint probabilities of a sequence $\{s_k\}$ may be determined easily from the joint probabilities of Section 1.

The block marked transmitter represents the signal generator, transmitting filter, and line of Figure 3. These functions may be lumped together without loss of generality. The transmitter is considered to put out a signal pulse or its negative in each half-bit cell depending on whether s_k is +1 or −1 in that half-bit cell. (The individual signal pulses extend beyond the boundaries of the half-bit cells, of course, because the system is bandwidth limited). In setting up a notation for the signal pulses, it is convenient to relocate the time origin so that t = 0 occurs at the middle of the first half of a bit cell. (See Figure 2 where t = 0 is at a bit cell boundary.) Among other (notational) advantages, this relocation allows the sample points to occur at $t = k\frac{T}{2}$, with k taking on integral values. The notation s(t) will be used for a (positive) pulse occurring in the zeroth half-bit cell. The pulse occurring in the k^{th} half-bit cell then becomes $s_k s(t - k\frac{T}{2})$, and an expression for the whole transmitted wave, S(t), is

$$S(t) = \sum_{k=-\infty}^{\infty} s_k s(t - k\frac{T}{2}) \tag{11}$$

To illustrate this notation, the data of Figure 2 are redrawn in Figure 5 for a typical s(t), logically characterized by s_k and with $s(t - k\frac{T}{2})$ shown in the zeroth and the eleventh bit cells.

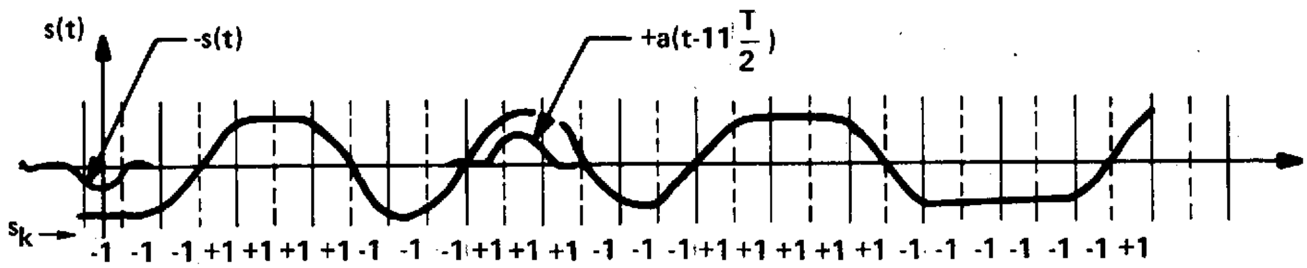


Figure 5. Typical Transmitted Wave (Figure 2 Data)

Note that symmetrical pulses have been drawn. Following the treatment in Bennett and Davey (Reference 1), the analysis assumes symmetrical pulses s(t) = s(-t) and works with nonrealizable filters. This does not entail any loss of generality since the equivalent system function could be approached arbitrarily closely using realizable filters with sufficient delay

The noise injected between the transmitter and the receiving filter is taken to be white with unilateral spectral density N_o. That is, if the noise voltage injected is N(t), its mean and autocorrelation functions are

$$E\,[N(t)] = 0$$

$$R\,(t-t') \equiv E\,[N(t)\,N(t')] = \frac{N_o}{2}\,\delta(t-t') \tag{12}$$

The impulse response function of the receiving filter is called h(t), and the filter output for a given input is designated by affixing a subscript 1 to the input function. Thus, corresponding to the single pulse input s(t), the wave train S(t) and the noise voltage N(t) are the outputs

$$
\begin{aligned}
s_1(t) &\equiv \int_{-\infty}^{\infty} dt'\, h(t-t')\, s(t') \\
S_1(t) &\equiv \int_{-\infty}^{\infty} dt'\, h(t-t')\, S(t') \\
N_1(t) &\equiv \int_{-\infty}^{\infty} dt'\, h(t-t')\, N(t')
\end{aligned}
\tag{13}
$$

It follows from Eq. (11) that

$$
S_1(t) = \sum_{k=-\infty}^{\infty} s_k\, s_1\left(t - k\,\frac{T}{2}\right) \tag{14}
$$

and from Eq. (12) that

$$
\begin{aligned}
&E\,[N_1(t)] = 0 \\
&R_1(t-t') \equiv E\,[N_1(t)\,N_1(t')] = \frac{N_o}{2} \int_{-\infty}^{\infty} dt'\, h(t-t'')\, h(t'-t'')
\end{aligned}
\tag{15}
$$

To summarize all this notation, the block diagram of Figure 4 is redrawn and the functions defined listed on the appropriate part of Figure 6.

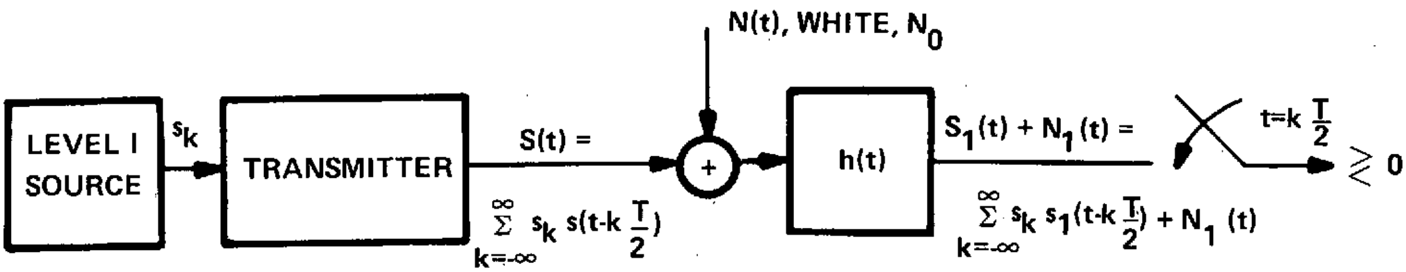


Figure 6. Notation Summary

It is convenient in some stages of the analysis to work in the frequency domain. Let x(t) represent any of the functions of time defined above. The Fourier transform of x(t) is called $\bar{x}(\omega)$ and is defined by the relations

$$\bar{x}(\omega) \equiv \int_{-\infty}^{\infty} dt\, e^{-i\omega t}\, x(t)$$
$$x(t) \equiv \int_{-\infty}^{\infty} \frac{d\omega}{2\pi}\, e^{i\omega t}\, \bar{x}(\omega) \qquad (16)$$

The Fourier transforms of Eq. (11) – (15) are

$$\bar{S}(\omega) = \bar{s}(\omega) \sum_{k=-\infty}^{\infty} s_k\, e^{-ik\frac{\omega T}{2}} \qquad (11')$$

$$\bar{R}(\omega) = \frac{N_o}{2} \qquad (12')$$

$$\bar{s}_1(\omega) = \bar{h}(\omega)\, \bar{s}(\omega) \qquad (13')$$

$$\bar{S}_1(\omega) = \bar{h}(\omega)\, \bar{S}(\omega)$$
$$\bar{S}_1(\omega) = \bar{s}_1(\omega) \sum_{k=-\infty}^{\infty} s_k\, e^{-ik\frac{\omega T}{2}} \qquad (14')$$

$$\bar{R}_1(\omega) = \frac{N_o}{2}\, |\bar{h}(\omega)|^2 \qquad (15')$$

$\bar{R}(\omega)$ and $\bar{R}_1(\omega)$ are, of course, the power spectral densities of N(t) and $N_1(t)$, respectively. $\bar{h}(\omega)$ is referred to as the receiving filter transfer function.

2.3 Optimum Signal Pulses and Filter

As mentioned before, the transmitted signal pulse s(t) and the receiving filter transfer function of the system to be analyzed are chosen to maximize the output signal-to-noise ratio at fixed transmitted power and to minimize intersymbol interference. Following Nyquist, the output single pulse spectrum $\bar{s}_1(\omega)$ is chosen to be a raised cosine of bandwidth 1/2T. That is,

$$\bar{s}_1(\omega) = \begin{cases} \frac{1}{2} VT \cos^2 \frac{\omega T}{4} & ; |\omega| < \frac{2\pi}{T} \\ 0 & ; |\omega| > \frac{2\pi}{T} \end{cases} \qquad (17')$$

or equivalently,

$$s_1(t) = \frac{1}{2} V \quad \frac{\sin 2\pi t/T}{2\pi t/T \, [1 - (2t/T)^2]} \tag{17}$$

This choice does not completely eliminate intersymbol interference with respect to samples taken in the middle of each half-bit cell (at $t = k\ T/2$); however, since

$$\begin{aligned} &s_1(0) = \frac{1}{2} V \ ; \quad s_1 \left(\pm \frac{T}{2}\right) = \frac{1}{4} V \ ; \\ &s_1 \left(k \frac{T}{2}\right) = 0; \quad k \neq 0, \pm 1 \end{aligned} \tag{18}$$

it prevents intersymbol interference except between neighboring samples. For a Level I system such as the laboratory system in which (1) the received wave train is squared up by sampling in the middle of each half-bit cell and comparing this value against a threshold, and (2) information is sent along a bandlimited channel at the Nyquist rate, the intersymbol interference cannot be reduced further.

Eq. (18) shows the meaning of the normalization constant V. If the nearest neighbors of a given half-bit cell both have pulses of the same sign, the nearest neighbor contributions add, and the sampled value is $\pm V$. If the nearest neighbors of the given half-bit cell have pulses of the opposite sign, their contributions subtract leaving the bit cell's sampled value of $\pm 1/2\,V$. (See the comments at the beginning of Section 1.3.) This normalization is different from that suggested by the labeling of Figure 2, which implies that $\pm V$ are the actual levels in the transmitted signal train. In the present convention, $\pm V$ are the levels (steady state after a long string of $+1^s$ or -1^s) after filtering by the receiving filter. This normalization is more convenient and, in any case, Section 2.5 shows how to express V in terms of both the average and the peak transmitted power.

With regard to the optimum receiving filter transfer function, Sunde has shown (Reference 1, Chapter 7) that, with respect to samples taken at the middle of the half-bit cells, the maximum signal-to-noise ratio (with a white noise input) is obtained by choosing

$$|\bar{h}(\omega)| \sim |\bar{s}(\omega)| \tag{19}$$

The normalization constant can be chosen so that the filter has unity gain at zero frequency, and the phase characteristic can be chosen for convenience. Note that if the normalization and phase were chosen so that $\bar{h}(\omega) = \bar{s}(\omega)^*$ the statement in the time domain would be $h(t) = s(-t)$. Thus it is seen that Sunde's result is just a generalization of the matched filter to the case where the signal pulses have nonvanishing values outside their own half-bit cells. The optimum receiving filter (to within a nonessential normalization and phase characteristic) is, therefore, just a filter matched to the transmitted signal pulse function $s(t)$.

From Eq. (13') and (19) it may be inferred that

$$s_1(\omega) \sim |h(\omega)|^2 \tag{20}$$

We therefore choose (see Eq. (17'))

$$\bar{h}(\omega) = \begin{cases} \cos\dfrac{\omega T}{4} & ; |\omega| < \dfrac{2\pi}{T} \\ 0 & ; |\omega| > \dfrac{2\pi}{T} \end{cases} \tag{21}$$

Eq. (13'), (17'), and (19) then give

$$\bar{s}(\omega) = \begin{cases} \dfrac{1}{2} VT \cos\dfrac{\omega T}{4} & ; |\omega| < \dfrac{2\pi}{T} \\ 0 & ; |\omega| > \dfrac{2\pi}{T} \end{cases} \tag{22'}$$

or equivalently

$$s(t) = \frac{1}{2} V \frac{4}{\pi} \frac{\cos 2\pi t/T}{[1 - (4t/T)^2]} \tag{22}$$

This completes the specification of the system. To summarize, raised cosine spectra pulses were selected for the output pulses out of the receiving filter to minimize intersymbol interference, and the receiving filter was matched to the transmitted pulse shape to maximize the sampled output signal-to-noise ratio. To list these results in a form for convenient reference, Figure 6 is redrawn and labeled accordingly in Figure 7.

2.4 Sampling Theorems

As indicated in Figure 7, the wave trains before and after filtering can be expressed in the forms

$$S(t) = \frac{1}{2} V \sum_{k=-\infty}^{\infty} s_k \frac{4}{\pi} \frac{\sin\frac{2\pi}{T}\left(t - k\frac{T}{2}\right)}{\left[1 - \left(\frac{4}{T}\left(t - k\frac{T}{2}\right)\right)^2\right]} \tag{23}$$

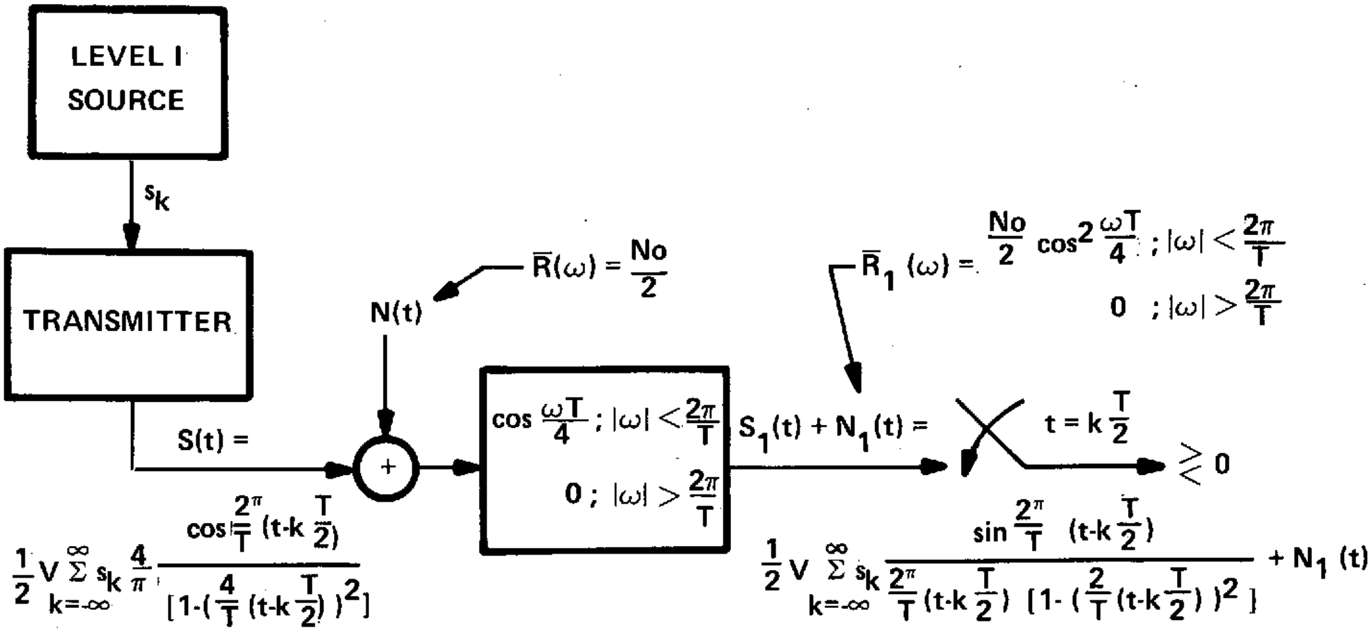


Figure 7. Section 2 Level I System

and

$$S_1(t) = \frac{1}{2} V \sum_{k=-\infty}^{\infty} s_k \frac{\sin \frac{2\pi}{T}\left(t - k\frac{T}{2}\right)}{\frac{2\pi}{T}\left(t - k\frac{T}{2}\right)\left[1 - \left(\frac{2}{T}\left(t - k\frac{T}{2}\right)\right)^2\right]} \tag{24}$$

In this subsection, a number of sampling theorems are derived to enable these series to be summed, yielding simple analytic expressions for S(t) and $S_1(t)$ whenever the series

$$\sum_{k=-\infty}^{\infty} s_k e^{ik\frac{\omega T}{2}}$$

can be summed. The development of these sampling theorems proceeds as follows. First, the sequence $\{s_k\}$ is represented by a discrete Fourier transform of minimum period T.

$$s_k = \int_{-2\pi/T}^{2\pi/T} \frac{d\omega}{2\pi} e^{-ik\frac{\omega T}{2}} \bar{f}(\omega) \tag{25}$$

$$\bar{f}(\omega) \equiv \frac{T}{2} \sum_{k=-\infty}^{\infty} s_k e^{ik\frac{\omega T}{2}}$$

Then the function f(t) is defined according to

$$f(t) \equiv \int_{-2\pi/T}^{2\pi/T} \frac{d\omega}{2\pi} e^{-i\omega t} \bar{f}(\omega) \tag{26}$$

f(t) is simply a (convenient) analog function that goes through the points s_k at $t = k\frac{T}{2}$ and contains no frequencies higher than 1/T. That is

$$f(k\frac{T}{2}) = s_k \tag{27}$$

It may be constructed by associating the values s_k with the middle of each half-bit cell and then drawing the smoothest curve through the points. (See Figure 8.)

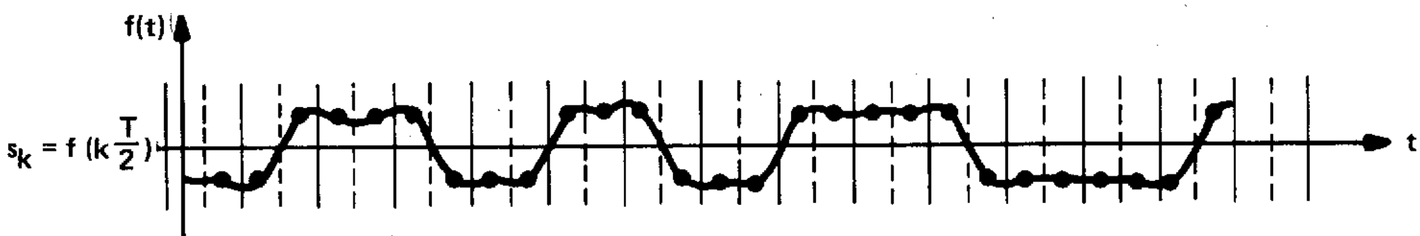


Figure 8. The Function f(t) (Figure 2 Data)

The usual sampling theorem may now be obtained by substituting Eq. (25) for $\bar{f}(\omega)$ into Eq. (26) and doing the ω integration. The result is

$$f(t) = \sum_{k=-\infty}^{\infty} s_k \frac{\sin \frac{2\pi}{T}(t - k\frac{T}{2})}{\frac{2\pi}{T}(t - k\frac{T}{2})} \tag{28}$$

Eq. (28) is usually thought of as an analytical representation of a function f(t) that contains no frequency components higher than 1/T in terms of its sample values, where the samples are spaced T/2 seconds apart. Here, it is regarded as the formula for an analog function containing no frequencies higher than 1/T that has been constructed to have the property

$$f(k\frac{T}{2}) = s_k$$

The next step is to express S(t) and $S_1(t)$ of Eq. (23) and (24) in terms of the function f(t) so constructed. To obtain a formula for S(t) in terms of f(t), note that (from Eq. (28))

$$f(t \pm \frac{T}{4}) = \pm \sum_{k=-\infty}^{\infty} s_k \frac{\cos \frac{2\pi}{T}(t - k\frac{T}{2})}{\frac{2\pi}{T}(t - k\frac{T}{2} \pm \frac{T}{4})} \tag{29}$$

After algebraic manipulation

$$f(t+\frac{T}{4}) + f(t-\frac{T}{4}) = \sum_{k=-\infty}^{\infty} s_k \frac{\cos \frac{2\pi}{T}(t - k\frac{T}{2})}{\left[1 - \left(\frac{4}{T}(t-k\frac{T}{2})\right)^2\right]} \tag{30}$$

Comparing with Eq. (23) yields

$$S(t) = \frac{1}{2} V \left[f(t + \frac{T}{4}) + f(t - \frac{T}{4})\right] \tag{31}$$

Similarly, to obtain a formula for $S_1(t)$ in terms of f(t), note that

$$f(t \pm \frac{T}{2}) = -\sum_{k=-\infty}^{\infty} s_k \frac{\sin \frac{2\pi}{T}(t - k\frac{T}{2})}{\frac{2\pi}{T}(t - k\frac{T}{2} \pm \frac{T}{2})} \tag{32}$$

After somewhat more algebra, the result is

$$\frac{1}{2}f(t + \frac{T}{2}) + f(t) + \frac{1}{2}f(t - \frac{T}{2}) = \sum_{k=-\infty}^{\infty} s_k \frac{\sin \frac{2\pi}{T}(t - k\frac{T}{2})}{\frac{2\pi}{T}(t-k\frac{T}{2})\left[1 - \left(\frac{2}{T}(t-k\frac{T}{2})\right)^2\right]} \tag{33}$$

Then comparing with Eq. (24)

$$S_1(t) = \frac{1}{2} V \left[\frac{1}{2}f(t+\frac{T}{2}) + f(t) + \frac{1}{2} f(t - \frac{T}{2})\right] \tag{34}$$

The sampling theorems just derived, namely Eq. (28), (30), and (33), are not directly applicable to the analysis of Manchester biphase. To obtain sampling theorems that are, Eq. (25) is used to define a function $f'(t)$ according to

$$f'(t) \equiv \frac{1}{2} \int_{-4\pi/T}^{4\pi/T} \frac{d\omega}{2\pi} e^{-i\omega t} \bar{f}(\omega) \tag{35}$$

which satisfies

$$f'(k\frac{T}{2}) = s_k$$

Thus $f'(t)$ is an analytic function that goes through the points s_k at $t = k\frac{T}{2}$, but it contains frequency components as high as 2/T according to the prescription of Eq. (35) used in conjunction with Eq. (25). The three sampling theorems that may be derived in terms of $f'(t)$ are

$$f'(t) = \sum_{k=-\infty}^{\infty} s_k \frac{\sin \frac{4\pi}{T}\left(t - k\frac{T}{2}\right)}{\frac{4\pi}{T}\left(t - k\frac{T}{2}\right)} \tag{37}$$

$$f'\left(t+\frac{T}{8}\right) + f'\left(t-\frac{T}{8}\right) = \sum_{k=-\infty}^{\infty} s_k \frac{4}{\pi} \frac{\cos\frac{4\pi}{T}\left(t - k\frac{T}{2}\right)}{\left[1 - \left(\frac{8}{T}\left(t - k\frac{T}{2}\right)\right)^2\right]} \tag{38}$$

$$\frac{1}{2}f'\left(t+\frac{T}{4}\right) + f'(t) + \frac{1}{2}f'\left(t-\frac{T}{4}\right) = \sum_{k=-\infty}^{\infty} s_k \frac{\sin\frac{4\pi}{T}\left(t - k\frac{T}{2}\right)}{\frac{4\pi}{T}\left(t-k\frac{T}{2}\right)\left[1 - \left(\frac{4}{T}\left(t-k\frac{T}{2}\right)\right)^2\right]} \tag{39}$$

2.5 Level I Wave Trains; Transmitted Power

Valuable insights into the nature of the Level I link may be obtained by using the sampling theorems of Section 2.4 to get analytic expressions for S(t) and $S_1(t)$ in various special cases. The simplest case to analyze is that in which the original binary source produces all 0's resulting in transitions in the Level I bit stream every 1-1/2 bits. For this case,

$$f(t),\ S(t),\ \frac{1}{3T}\int_0^{3T} dt\, S(t)^2,\ \text{and}\ S_1(t)$$

may then be computed.

CASE 1 - Transitions every 1-1/2 bits.

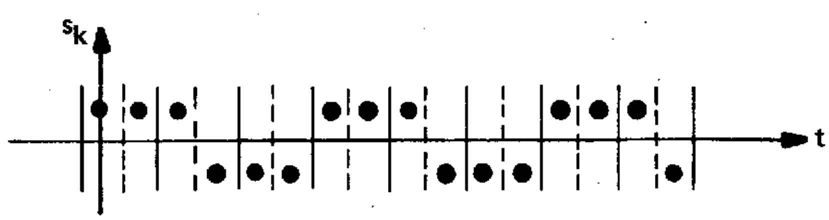


Figure 9. Case 1 Bit Stream

For this case,

$$s_{3\ell} = s_{3\ell+1} = s_{3\ell+2} = (-1)^{\ell} \; ; \ell = \ldots, -1, 0, 1,], \ldots\ldots \tag{40}$$

Therefore

$$\bar{f}(\omega) = \frac{T}{2} \sum_{k=-\infty}^{\infty} s_k \, e^{ik\frac{\omega T}{2}} = \frac{T}{2} \sum_{k=-\infty}^{\infty} (-1)^{\ell} \left[e^{i3\ell\frac{\omega T}{2}} + e^{i(3\ell+1)\frac{\omega T}{2}} + e^{i(3\ell+2)\frac{\omega T}{2}} \right]$$
$$= \frac{T}{2} \left(1 + e^{i\frac{\omega T}{2}} + e^{i\omega T} \right) \sum_{\ell=-\infty}^{\infty} (-1)^{\ell} e^{i\ell\frac{3\omega T}{2}} \tag{41}$$

The series $\sum_{\ell=-\infty}^{\infty} (-1)^{\ell} e^{i\ell\frac{3\omega T}{2}}$ may be summed according to the following method:

$$\frac{3T}{2} \sum_{\ell=-\infty}^{\infty} (-1)^{\ell} e^{i\ell\frac{3\omega T}{2}} = \lim_{\substack{N\to\infty \\ N \text{ odd}}} \frac{3T}{2} \sum_{\ell=-\left(\frac{N-1}{2}\right)}^{\frac{N-1}{2}} e^{i\ell\left(\frac{3\omega T}{2} + \pi\right)}$$

$$= \lim_{\substack{N\to\infty \\ N \text{ odd}}} \frac{3T}{2} \, e^{-i\left(\frac{N-1}{2}\right)\left(\frac{3\omega T}{2} + \pi\right)} \left[1 + e^{i\left(\frac{3\omega T}{2} + \pi\right)} + \ldots + e^{i(N-1)\left(\frac{3\omega T}{2} + \pi\right)} \right]$$

$$= \lim_{\substack{N\to\infty \\ N \text{ odd}}} \frac{3T}{2} \, e^{-i\left(\frac{N-1}{2}\right)\left(\frac{3\omega T}{2} + \pi\right)} \frac{1 - e^{iN\left(\frac{3\omega T}{2} + \pi\right)}}{1 - e^{i\left(\frac{3\omega T}{2} + \pi\right)}} \tag{42}$$

$$= \lim_{\substack{N\to\infty \\ N \text{ odd}}} \frac{3T}{2} \frac{\sin N\left(\frac{3\omega T}{4} + \pi/2\right)}{\sin\left(\frac{3\omega T}{4} + \pi/2\right)}$$

The diffraction grating amplitude function $3T/2 \sin N (3\omega T/4 + \pi/2)$ (x) $[\sin(3\omega T/4 + \pi/2)]^{-1}$ has principal maxima at $3\omega T/4 + \pi/2 = n\pi$, where n is an integer. In terms of ω, the maxima occur at $\omega = (2n-1) \, 2\pi/3T$. In the limit as N goes to infinity, the function becomes a series of delta functions at these points. The normalization is determined by the normalization integral

$$\lim_{N \to \infty} \int_{(2n-1)\frac{2\pi}{3T} - \epsilon}^{(2n-1)\frac{2\pi}{3T} + \epsilon} d\omega \, \frac{3T}{2} \, \frac{\sin N\left(\frac{3\omega T}{4} + \pi/2\right)}{\sin\left(\frac{3\omega T}{4} + \frac{\pi}{2}\right)} = 2\pi \tag{43}$$

Therefore, the result for the series is

$$\frac{3T}{2} \sum_{\ell = -\infty}^{\infty} (-1)^{\ell} \, e^{i\ell \frac{3\omega T}{2}} = 2\pi \sum_{n=-\infty}^{\infty} \delta\left[\omega - (2n-1)\frac{2\pi}{3T}\right] \tag{44}$$

which, with Eq. (41), gives the result for the discrete Fourier transform of s_k; namely

$$\begin{aligned} \bar{f}(\omega) &= \frac{1}{3}\left(1 + e^{i\frac{\omega T}{2}} + e^{i\omega T}\right) 2\pi \sum_{n=-\infty}^{\infty} \delta\left[\omega - (2n-1)\frac{2\pi}{3T}\right] \\ &= 2\pi \sum_{n=-\infty}^{\infty} \frac{1}{3}\left(1 + e^{i(2n-1)\pi/3} + e^{i(2n-1)\frac{2\pi}{3}}\right) \delta\left[\omega - (2n-1)\frac{2\pi}{3T}\right] \end{aligned} \tag{45}$$

Using the relation

$$1 + e^{i(2n-1)\pi/3} + e^{i(2n-1)2\pi/3} = \frac{1 - e^{i(2n-1)\pi}}{1 - e^{i(2n-1)\pi/3}} = \frac{2}{1 - e^{i(2n-1)\pi/3}} \tag{46}$$

Eq. (45) can be written in the form

$$\bar{f}(\omega) = 2\pi \sum_{n=-\infty}^{\infty} \frac{2/3}{1 - e^{i(2n-1)\pi/3}} \, \delta\left[\omega - (2n-1)\frac{2\pi}{3T}\right] \tag{47}$$

The next step is the computation of f(t) using Eq. (26). The result is

$$f(t) = \frac{2}{3} \, \frac{\sin\left(\frac{2\pi t}{3T} + \frac{\pi}{6}\right)}{\sin \pi/6} + \frac{1}{3} \, \frac{\sin\left(\frac{2\pi t}{T} + \pi/2\right)}{\sin \pi/2} \tag{48}$$

The factor multiplying the second term is 1/3 rather than 2/3 because the delta function has only been integrated halfway into. That this is the right thing to do may be checked using summability procedures similar to those used in deriving Eq. (44).

Finally, Eq. (31) and (33) may be used to determine S(t) and $S_1(t)$. The results are

$$S(t) = \frac{2}{3} V \cot \pi/6 \sin\left(\frac{2\pi t}{3T} + \frac{\pi}{6}\right) \tag{49}$$

$$S_1(t) = \frac{2}{3} V \cot \pi/6 \cos \pi/6 \sin\left(\frac{2\pi t}{3T} + \frac{\pi}{6}\right) \tag{50}$$

The average power in the wave train of Eq. (49) is

$$\frac{1}{3T} \int_0^{3T} dt\, S(t)^2 = \frac{2}{3} V^2 \tag{51}$$

In cases 2-6 below, similar results are summarized for transitions every two bit cells through transitions every four bit cells (Figures 10 - 14).

CASE 2 - Transitions every 2 bits.

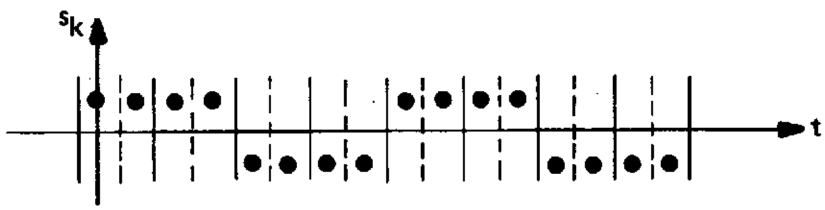


Figure 10. Case 2 Bit Stream

$$s_{4\ell} = s_{4\ell+1} = s_{4\ell+2} = (-1)^{\ell} \quad ; \quad \ell = \ldots\ldots, -1, 0, 1, 2, \ldots \tag{52}$$

$$f(t) = \frac{1}{2} \frac{\sin\left(\frac{\pi t}{2T} + \pi/8\right)}{\sin \pi/8} + \frac{1}{2} \frac{\sin\left(\frac{3\pi t}{2T} + 3\pi/8\right)}{\sin 3\pi/8} \tag{53}$$

$$S(t) = \frac{1}{2} V \left[\cot \pi/8 \sin\left(\frac{\pi t}{2T} + \pi/8\right) + \cot \frac{3\pi}{8} \sin\left(\frac{3\pi t}{2T} + \frac{3\pi}{8}\right)\right] \tag{54}$$

$$\frac{1}{4T}\int_0^{4T} dt\, S(t)^2 = \frac{3}{4}\, V^2 \tag{55}$$

$$S_1(t) = \frac{1}{2} V \left[\cot\frac{\pi}{8}\cos\frac{\pi}{8}\sin\left(\frac{\pi t}{2T}+\frac{\pi}{8}\right) + \cot\frac{3\pi}{8}\cos\frac{3\pi}{8}\sin\left(\frac{3\pi t}{2T}+\frac{3\pi}{8}\right)\right] \tag{56}$$

For Cases 3-6, only the average transmitted power and $S_1(t)$ are listed, since S(t) and f(t) are rather simply related to $S_1(t)$ as in Cases 1 and 2.

CASE 3 - Transitions every 2-1/2 bits.

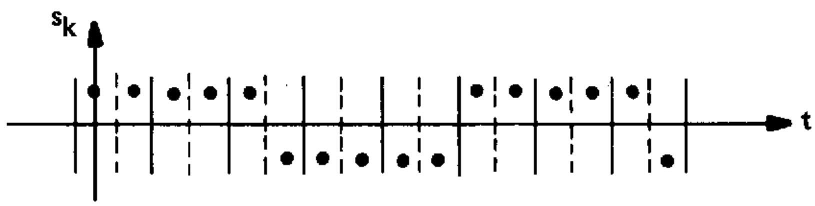


Figure 11. Case 3 Bit Stream

$$s_{5\ell} = s_{5\ell+1} = s_{5\ell+2} = s_{5\ell+3} = s_{5\ell+4} = (-1)^\ell; \quad \ell = \ldots, -1, 0, 1, 2, \ldots \tag{57}$$

$$\frac{1}{5T}\int_0^{5T} dt\, S(t)^2 = \frac{4}{5} V^2 \tag{58}$$

$$S_1(t) = \frac{2}{5} V \left[\cot\frac{\pi}{10}\cos\frac{\pi}{10}\sin\left(\frac{2\pi t}{5T}+\frac{\pi}{10}\right) + \cot\frac{3\pi}{10}\cos\frac{3\pi}{10}\sin\left(\frac{6\pi t}{5T}+\frac{3\pi}{10}\right)\right] \tag{59}$$

CASE 4 - Transitions every 3 bits.

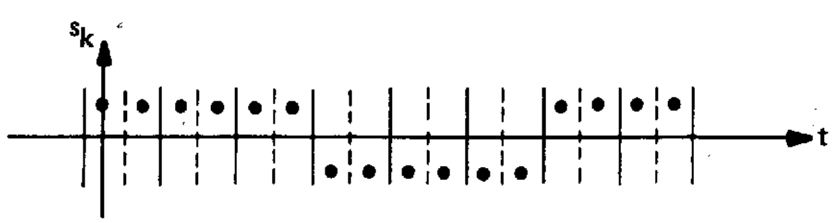


Figure 12. Case 4 Bit Stream

$$s_{6\ell} = \ldots = s_{6\ell+5} = (-1)^{\ell} \quad ; \quad \ell = \ldots, -1, 0, 1, 2, \ldots \tag{60}$$

$$\frac{1}{6T} \int_0^{6T} dt \; S(t)^2 = \frac{5}{6} V^2 \tag{61}$$

$$\begin{aligned} S_1(t) = \frac{1}{3} V \Bigg[& \cot\frac{\pi}{12} \cos\frac{\pi}{12} \sin\left(\frac{\pi t}{3T} + \pi/12\right) + \cot\frac{\pi}{4} \cos\frac{\pi}{4} \sin\left(\frac{\pi t}{T} + \pi/4\right) \\ & + \cot\frac{5\pi}{12} \cos\frac{5\pi}{12} \sin\left(\frac{5\pi t}{3T} + \frac{5\pi}{12}\right) \Bigg] \end{aligned} \tag{62}$$

CASE 5 - Transition every 3-1/2 bits

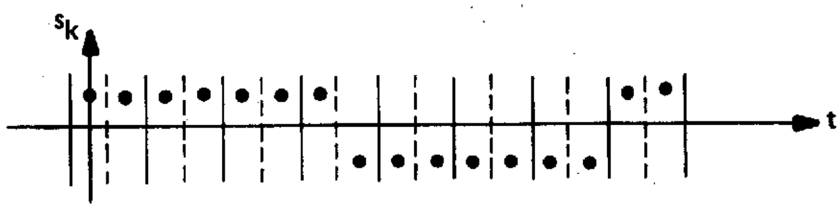


Figure 13. Case 5 Bit Stream

$$\frac{1}{7T} \int_0^{7T} dt \; S(t)^2 = \frac{6}{7} V^2 \tag{63}$$

$$\begin{aligned} S_1(t) = \frac{2}{7} V \Bigg[& \cot\frac{\pi}{14} \cos\frac{\pi}{14} \sin\left(\frac{2\pi t}{7T} + \frac{\pi}{14}\right) + \cot\frac{3\pi}{14} \cos\frac{3\pi}{14} \sin\left(\frac{6\pi t}{7T} + \frac{3\pi}{14}\right) \\ & + \cot\frac{5\pi}{14} \cos\frac{5\pi}{14} \sin\left(\frac{10\pi t}{7T} + \frac{5\pi}{14}\right) \Bigg] \end{aligned} \tag{64}$$

CASE 6 - Transition every four bits

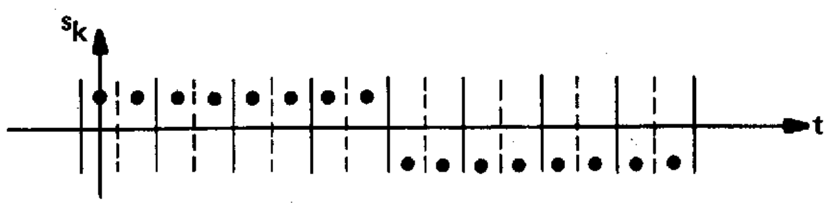


Figure 14. Case 6 Bit Stream

$$\frac{1}{8T} \int_0^{8T} dt\ S(t)^2 = \frac{7}{8} V^2 \tag{65}$$

$$S_1(t) = \frac{1}{4} V \left[\cot\frac{\pi}{16} \cos\frac{\pi}{16} \sin\left(\frac{\pi t}{4T} + \frac{\pi}{16}\right) + \cot\frac{3\pi}{16} \cos\frac{3\pi}{16} \sin\left(\frac{3\pi t}{4T} + \frac{3\pi}{16}\right) \right.$$
$$\left. + \cot\frac{5\pi}{16} \cos\frac{5\pi}{16} \sin\left(\frac{5\pi t}{4T} + \frac{5\pi}{16}\right) + \cot\frac{7\pi}{16} \cos\frac{7\pi}{16} \sin\left(\frac{7\pi t}{4T} + \frac{7\pi}{16}\right) \right] \tag{66}$$

On the basis of the above results for the transmitted power, it is clear that the average transmitted power averaged over all possible Level I wave trains is between 2/3 V^2 and 7/8 V^2. A straight average of these two values gives 0.771 V^2. The computer simulation described in Section C computes the average power to be 0.775 V^2. To eliminate lengthy decimal fractions from the results, the formula relating V^2 to the average transmitted power P_{AV} is taken to be

$$P_{AV} \equiv \frac{3}{4} V^2 \tag{67}$$

This value differs from 0.775 V^2 by only 0.14 dB, which is negligible in the numerical calculations of Section D.

The peak transmitted signal voltage required is that in Case 1 above. From Eq. (49), it is 2/3 V cot $\pi/16 = 2/\sqrt{3}$ V. The peak transmitted power required is therefore

$$P_{PK} = \frac{4}{3} V^2 = \frac{16}{9} P_{AV} \tag{68}$$

2.6 Level I Bit Error Rate

(The characteristics of the system being considered are listed in Figure 7.) Because of the structure of the raised cosine spectrum pulses, the sampler measures the sample values $\pm V$ when the sampled half-bit cell is not adjacent to a transition and $\pm 1/2V$ when it is. As was shown in Section 1.3, the samples are $\pm V$ 4/7 of the time and $\pm 1/2$ V 3/7 of the time. The probability of making an error in comparing against a zero threshold to square up the filtered wave train is, therefore, 4/7 of the probability that the sampled noise voltage will exceed +V volts when the received signal voltage was +V, etc., plus 3/7 of the probability that the noise voltage will exceed 1/2 V, etc. The probability that a Gaussian distributed noise voltage of zero mean and variance σ^2 will exceed a voltage level +V is given by the standard formula

$$\Pr(N_1 > V) = \frac{1}{2}\,\mathrm{erfc}\left(\frac{V}{\sigma\sqrt{2}}\right) \tag{69}$$

where

$$\mathrm{erfc}\,x \equiv \frac{2}{\sqrt{\pi}} \int_x^\infty d\xi\, e^{-\xi^2}$$

In the present case

$$\sigma^2 = \int_{-\infty}^{\infty} \frac{d\omega}{2\pi}\, \overline{R}_1(\omega) = \frac{N_o}{4\pi} \int_{-2\pi/T}^{2\pi/T} d\omega \cos^2 \frac{\omega T}{4} = N_o/2T \tag{70}$$

The formula for the probability of making an error in determining the level of a given half-bit chosen at random is therefore

$$\Pr(\epsilon) = \frac{4}{7} \cdot \frac{1}{2}\,\mathrm{erfc}\left(\frac{V}{\sqrt{N_o/T}}\right) + \frac{3}{7} \cdot \frac{1}{2}\,\mathrm{erfc}\left(\frac{\frac{1}{2}V}{\sqrt{N_o/T}}\right) \tag{71}$$

Eq. (67) and (68) may be used to express this result in terms of either the average or the peak transmitted power. Thus

$$\Pr(\epsilon) = \frac{4}{7} \cdot \frac{1}{2}\,\mathrm{erfc}\left(\sqrt{\frac{4}{3}\,\frac{P_{AV}}{N_o/T}}\right) + \frac{3}{7} \cdot \frac{1}{2}\,\mathrm{erfc}\left(\sqrt{\frac{1}{3}\,\frac{P_{AV}}{N_o/T}}\right) \tag{72}$$

or

$$\Pr(\epsilon) = \frac{4}{7} \cdot \frac{1}{2}\,\mathrm{erfc}\left(\sqrt{\frac{3}{4}\,\frac{P_{PK}}{N_o/T}}\right) + \frac{3}{7} \cdot \frac{1}{2}\,\mathrm{erfc}\left(\sqrt{\frac{3}{16}\,\frac{P_{PK}}{N_o/T}}\right) \tag{73}$$

The bit error rate (BER) is the probabilityof an information bit being in error after Level I decoding. It may be obtained from $\Pr(\epsilon)$ by multiplying by 2 to convert from half bits to bits, and by multiplying by the number of decoded bits in error that result from one half-bit being in error (which will be called α). Thus

$$\mathrm{BER} = 2\,\alpha\,\Pr(\epsilon) \tag{74}$$

Arguments are presented below which suggest that

$$\alpha \approx 2 \tag{75}$$

(The actual value of α is correctly taken into account in the simulation.)

The complementary error functions in Eq. (72) or (73) are very rapidly decreasing functions of the arguments under the square roots. In all situations of interest, the first terms of Eq. (72) and (73) are completely negligible in comparison with the second terms. This implies that nearly all of the half-bit errors result from sampling the voltages $\pm 1/2V$; that is, in sampling half-bit cells that are adjacent to transitions. Thus, the half-bit errors almost always result in the shifting of a transition just one half-bit cell away from its correct position. Except in the cases where transitions are so close together that a shift in one results in a faulty interpretation of the other, a half-bit error that results in shifting a transition within the same bit cell (edge to mid, or mid to edge) results in one bit being in error in the decoded bit stream. A half-bit error that results in shifting the transition into a neighboring bit cell results in three bits in the decoded stream being in error. Thus on the average, neglecting the complicated cases, a single half-bit in error results in two bit errors in the Level I decoded bit stream.

3.0 DIGITAL SIMULATION

3.1 Overall Simulation

An overall block diagram of the digital simulation used to support this study is given in Figure 15. As illustrated in the figure, the simulation has been divided into three distinct program steps. They are:

1. Encoding Program
2. Transmission, Receiving and Detection Program
3. Decoding and Comparison Program.

3.2 Encoding Program

The purpose of the encoding program is to generate a magnetic tape containing the Level I coding of a random sequence of binary data. In addition, the output tape contains the original binary data. This data will be used to determine the number of errors made in the transmission, receiving, detection, and decoding process.

A simplified flow diagram of the Encoding Program is given in Figure 16A, and a FORTRAN listing of the program is given in Figure 16B.

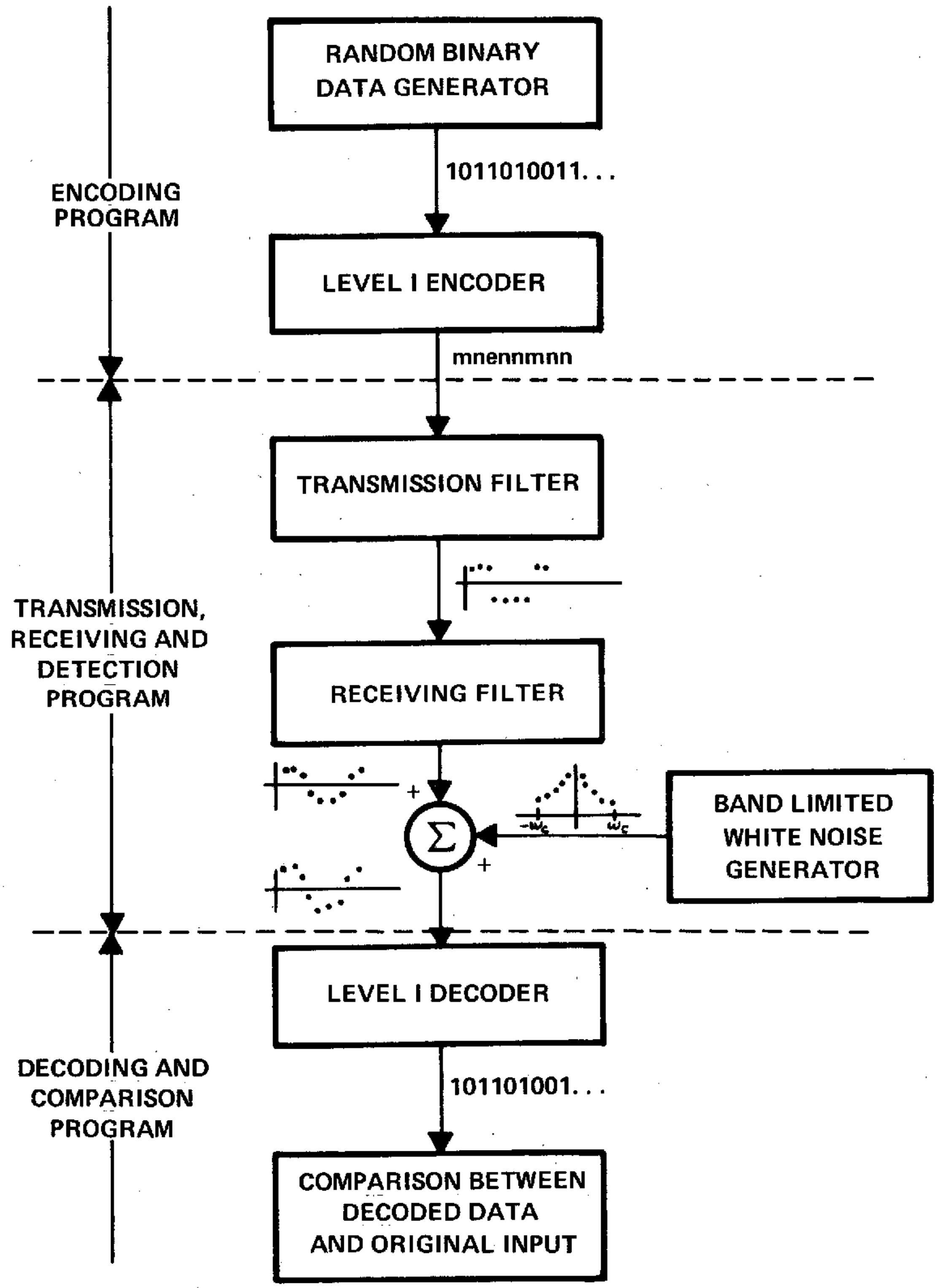


Figure 15. Overall Simulation Block Diagram

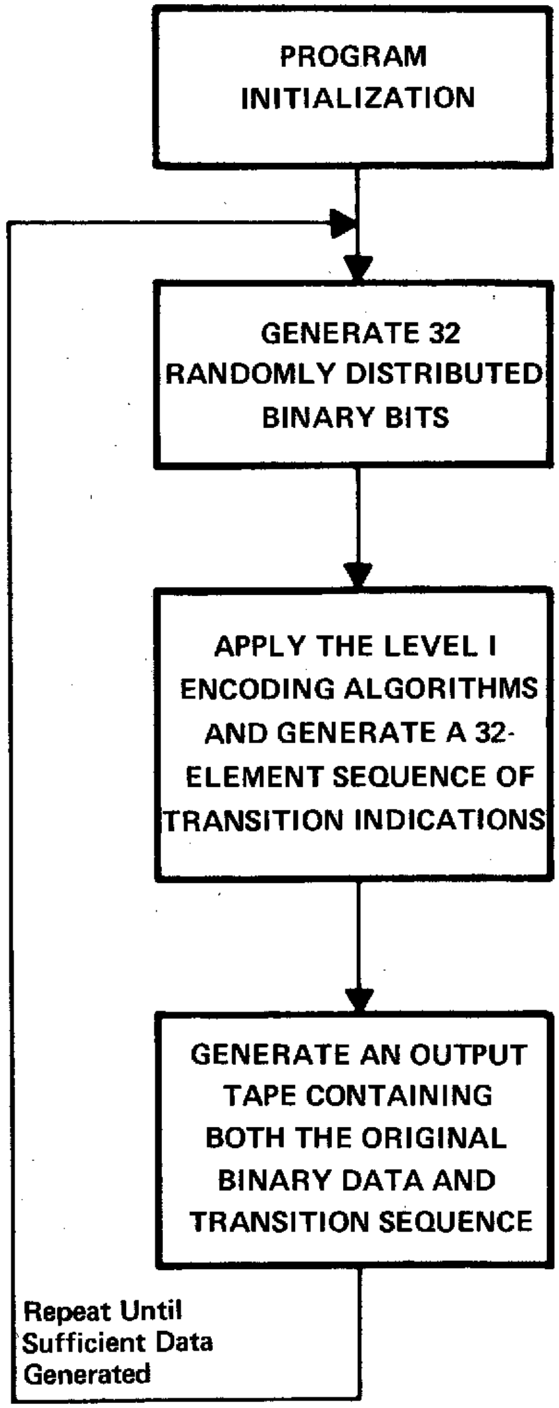


Figure 16A. Program to Generate a Sequence of Random Binary Bits and Perform Level I Encoding Function

```
      REAL*4 MSG(15)
      DATA MSG/' MID BIT TRANSITION ',' NO TRANSITION ',' EDGE TR
     1ANSITION '/
      INTEGER DATA, BITCOUNT, SHIFTCNT, DATATEMP, BITS, DATABITS,SEVEN
      INTEGER BITOUT
      INTEGER BITOUTP
      DATA BITS/8ZE0000000/
      DATA SEVEN/7/
      DIMENSION LEVEL1(64), LEVELOUT(64)
      RECNT = 325
      IX= 84637
      CALL RANDOM( IX, YFL )
      IDATA = IX
      BITCOUNT = 1
      J = 1
      DATA = IDATA
      DATABITS = IAND( ISA( IAND( DATA, BITS ), -29), SEVEN )
      DATA = ISA( DATA, +1 ) ; SHIFTCNT =1
    2 IF( DATABITS.GE.6 .OR. DATABITS.EQ.0 ) LEVEL1(BITCOUNT) = +1 ;
     1 GO TO 3
      IF( DATABITS.GE.4 ) LEVEL1(BITCOUNT) = -1 ; GO TO 3
      LEVEL1(BITCOUNT) = 0
    3 DATABITS = IAND( ISA( IAND( DATA, BITS ), -29), SEVEN )
      DATA = ISA( DATA, +1 ) ; SHIFTCNT = SHIFTCNT + 1
      IF( SHIFTCNT.GE.29 ) GO TO 110
    4 BITCOUNT = BITCOUNT + 1
      IF( LEVEL1(BITCOUNT-1).EQ.0 ) GO TO 2
      IF( LEVEL1(BITCOUNT-1).EQ.-1 ) LEVEL1(BITCOUNT) = 0 ; GO TO 3
    6 IF( DATABITS.GE.4 ) LEVEL1(BITCOUNT) = 0 ; GO TO 3
      IF( DATABITS.LE.1) LEVEL1(BITCOUNT) = -1 ; GO TO 3
      WRITE( 6,200 ) ; STOP
  200 FORMAT(T10,'AN UNACCOUNTED FOR BIT PATTERN HAS OCCURED')
  110 IF( SHIFTCNT .GT. 32 ) GO TO 121
      IF( SHIFTCNT .EQ. 32 ) GO TO 120
      IF( SHIFTCNT.GT.29 ) GO TO 4
  100 CALL RANDOM( IX, YFL )
      IDATATMP = IX
      DATATEMP = IAND( IDATATMP, BITS )
  111 DATATEMP = IAND( ISA( DATATEMP, -3 ), 8Z1C000000 )
      DATA = IOR( DATA, DATATEMP )
      GO TO 4
  120 WRITE( 6,202 ) IDATA ; WRITE( 7 ) IDATA
      IDATA = IDATATMP ; DATA = IDATA
      GO TO 4
  121 WRITE( 7 ) ( LEVEL1(I),I=1,32)
X     WRITE( 6,201 ) ((MSG((LEVEL1(I)+1)*5+K),K=1,5),I=1,32)
      RECORD = RECORD + 1 ; SHIFTCNT = 1 ; BITCOUNT = 1
      IF( RECORD.EQ.RECNT ) END FILE 7 ; REWIND 7 ; STOP
      CALL ANYPSON( 300S,1 )
  301 CONTINUE
      CALL ANYPSON( 302S,3 )
      IF( LEVEL1(32) .EQ. 0 ) GO TO 2
      IF( LEVEL1(32).EQ.-1 ) LEVEL1(1) = 0 ; GO TO 3
      GO TO 6
  300 OUTPUT(102) RECORD
      GO TO 301
  302 OUTPUT(102) RECORD
      ENDFILE 7 ; REWIND 7
      STOP
  201 FORMAT( T5,6(5A4))
  202 FORMAT(T10,'RANDOM DATA INPUT STREAM',5X,Z8/)
      END
```

Figure 16B. FORTRAN Listing of the Encoding Program

3.3 Transmission, Receiving and Detection Program

This program represents the main thrust of the simulation effort. As indicated by the title, this program simulates the transmission process, receiving process, and detection procedure. The resultant output is a magnetic tape containing the detected Level I coded pulse train and the original binary data which it represents.

While the transmission filter is explicitly defined in terms of its impulse response, the receiving filter is implicitly defined by the impulse of the "overall" system (transmission filter - receiving filter).

The first step in simulating the various system outputs is to convert the input transition sequence to a discrete pulse train of four pulses per bit cell, i.e., leading edge, one-quarter, half, and three-quarter points of the bit cell. This discrete pulse train is then convolved with the impulse response representing the transmission filter. This output is used to define the average transmitted power.

The output of the overall system (receiving filter output) is defined by convolving the discrete pulse train with the impulse response of the overall system. As part of the simulation of the overall system output, it is necessary to account for the effect of noise. This is accomplished by generating band-limited white noise with zero mean and variance σ^2 (as prescribed in Sections 2.2 and 2.6) and adding it to the already generated system output. This forms the overall system output to be used by the various detection schemes.

The output representing the one-quarter and three-quarter points of each bit cell are now used as input to the detection procedures. Two procedures are suggested in Sections 2.1 and 4.3 and a simplified description follows:

1. Single Point. This procedure issued a ±1 for each point depending on whether the amplitude is greater than or less than zero. The polarity of these two samples are now compared with the last previously generated data and a decision made as to whether an edge, mid bit, or no transition has occurred.

2. Four Point. The second procedure forms two weighted values utilizing the present samples and the three immediately previous samples. The absolute value of these weighted values are then compared to a threshold. For those values which exceed the threshold, the local maximum is determined, defining the point at which a transition has occurred.

An output tape containing both the decoded transition sequence and the original binary data is now generated. This output tape will be processed by another program to determine how many errors were made. A simplified block diagram of this program is given in Figure 17A and its corresponding FORTRAN listing is shown in Figure 17B.

3.4 Decoding and Comparison Program

This program takes the transition sequence generated by the transmission, receiving, and detection program and regenerates a binary data stream. This resulting data stream is then compared with the original binary data in order to determine how many errors were made as a result of the simulated system.

Figure 18A is a simplified block diagram of this program, and Figure 18B contains the FORTRAN listing.

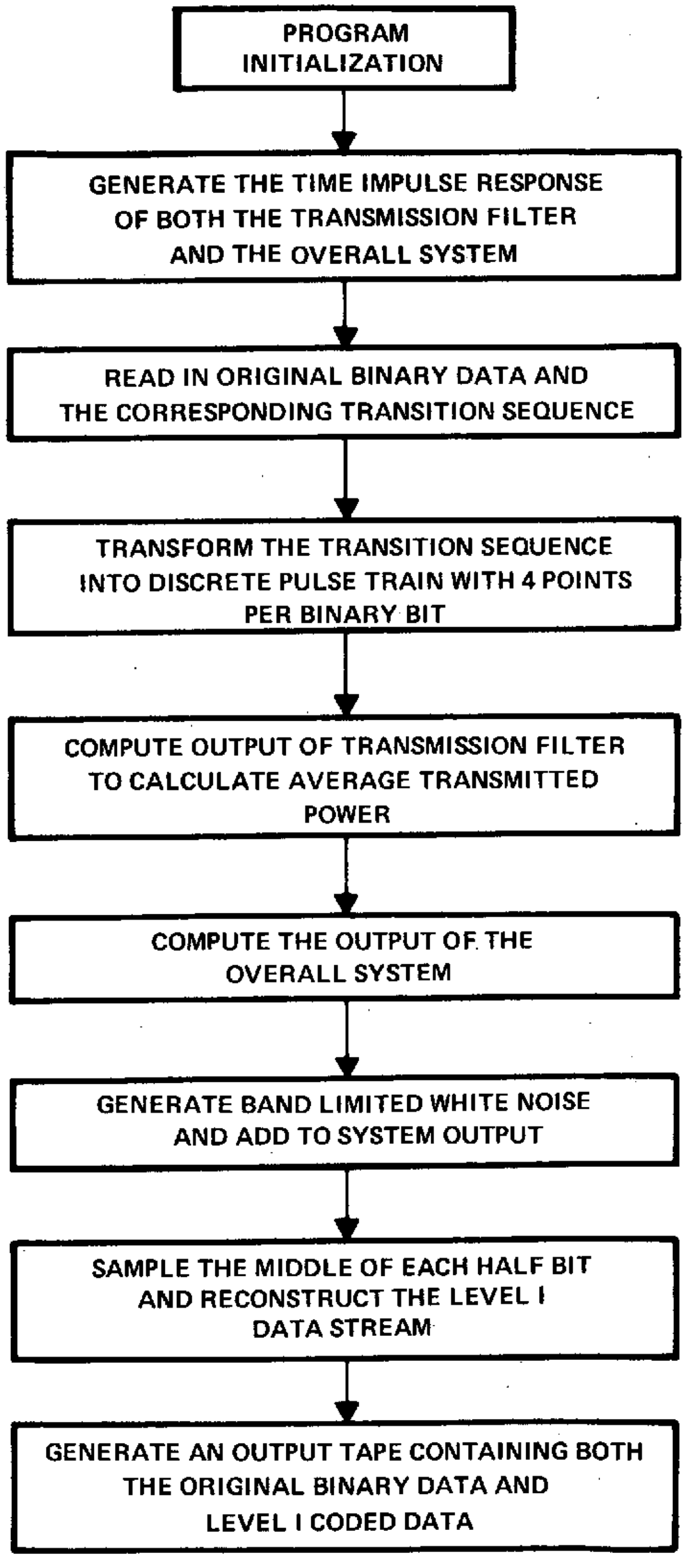


Figure 17A. Simulation of the Transmission, Receiving and Detection Process

```
      INTEGER DATA, BITØNE, DATABITS, R
      DIMENSIØN X(41), DATAØUT(12), H(41), P(2), DA(6), SIG(6),
     1 DATAØUT1(12),  SCRIPTS(41), CAPS(41), Q(2)
     2 , LEVELIN(32), QK(2)
      DIMENSIØN LEVEL1(128), LEVEL2(128), LEVEL3(128), LEVEL4(64),
     * LEVEL5(64), LEVEL6(64)
      DATA BITØNE/8Z80000000/
      NAMELIST SIGMA, THRESH2, THRESHDF
      INØISE = 49323
      IX = 84637 ; R = 9 ; BITCNT = 0
      DIFFP = 0.0 ; LEVEL3P = 1.0
      LEVEL2P = 1.0 ; DIFFP1 = 0.0
      THRESH2 = 2.5 ; THRESHDF = 1.25
      ISIGN2 = 1
      DØ 60 I = 1,41
   60 SCRIPTS(I) = 0.0
      SCRIPTS(21) = 4/3.1415927
      SCRIPTS(20) = SCRIPTS(22) = 1.0
      DØ 61 I = 2,20,2
      SCRIPTS(21-I) = 4*(-1)**(I/2)/(1-I**2)/3.1415927
   61 SCRIPTS(21+I) = SCRIPTS(21-I)
      DØ 62 I = 1,41
   62 H(I) = 0.0
      H(21) = 1.0
      H(20) = H(22) = 8/(3.1415927*3)
      H(19) = H(23) = 0.5
      DØ 63 I = 3,19,2
      H(21-I) = (-1)**(I/2. - 1/2.)*8/(4-I**2)/3.1415927/3
   63 H(21+I) = H(21-I)
      INPUT
      ØUTPUT SIGMA, THRESH2, THRESHDF
      XMEAN = 0.0
X     CALL STRIPULSE(20)
X     CALL STRIPSPEED(10)
      DØ 30 I = 1,6
   30 DA(I) = 1.0
X     CALL STRIPLØT(DA)
      DØ 31 I = 1,6
   31 DA(I) = -1.0
X     CALL STRIPLØT(DA)
      DØ 32 I = 1,6
   32 DA(I) = 0.0
X     CALL STRIPULSE(150)
X     CALL STRIPLØT(DA)
X     CALL STRIPULSE(10)
      PLAST = 1.0 ; SIGNLAST = 1.0
      DØ 10 N = 1,325
      READ( 7, END=100 ) IDATA ; READ(7)(LEVELIN(J),J=1,32)
      DØ 9 J = 1,32
      IF( LEVELIN(J).LT.0 ) P(1)= PLAST ; P(2) =-P(1)
      IF( LEVELIN(J).EQ.0 ) P(1) = P(2) = PLAST
      IF( LEVELIN(J).GT.0) P(1) = P(2) = -PLAST
      PLAST = P(2)
      DA(1) = LEVELIN(J)/2.0
    1 DØ 2 M = -20,20,1
    2 X(M+21) = X(M+21) + P(1)*H(M+21)
      DATAØUT(R) = X(1)  ; DATAØUT(R+1) = X(2)
      DØ 42 MM = -20,20,1
   42 CAPS(MM+21) = CAPS(MM+21) + P(1)*SCRIPTS(MM+21)
```

Figure 17B (Sheet 1 of 4). FORTRAN Listing of the Transmission, Receiving and Detection Process

ORIGINAL PAGE IS
OF POOR QUALITY

```
      PØWER = CAPS(1)**2 + CAPS(2)**2 + PØWER
      NP = NP + 2
      DØ 3 K = -20,18,1
    3 X(K+21) = X(K+23)
      X(40) = X(41) = 0.0
      DØ 43 KK = -20,18,1
   43 CAPS(KK+21) = CAPS(KK+23)
      CAPS(40) = CAPS(41) = 0.0
      DØ 4 M = -20,20,1
    4 X(M+21) = X(M+21) + P(2)*H(M+21)
      DATAØUT(R+2) = X(1) ; DATAØUT(R+3) = X(2)
      DØ 44 MM = -20,20,1
   44 CAPS(MM+21) = CAPS(MM+21) + P(2)*SCRIPTS(MM+21)
      PØWER = CAPS(1)**2 + CAPS(2)**2 + PØWER
      NP = NP + 2
      DØ 5 K = -20,18,1
    5 X(K+21) = X(K+23)
      DØ 45 KK = -20,18,1
   45 CAPS(KK+21) = CAPS(KK+23)
      CAPS(40) = CAPS(41) = 0.0
      X(40) = X(41) = 0.0
      DA(2) = P(1)/2.0
      DA(3) = DATAØUT(R)/2.0
      CALL GAUSSØM( INØISE, SIGMA, XMEAN, SNØISE )
      DATAØUT1(R) = DATAØUT(R) + SNØISE
      CALL GAUSSØM( INØISE, SIGMA, XMEAN, SNØISE )
      DATAØUT1(R+1) = DATAØUT(R+1) + SNØISE
      CALL GAUSSØM( INØISE, SIGMA, XMEAN, SNØISE )
      DATAØUT1(R+2) = DATAØUT(R+2) + SNØISE
      CALL GAUSSØM( INØISE, SIGMA, XMEAN, SNØISE )
      DATAØUT1(R+3) = DATAØUT(R+3) + SNØISE
      DA(4) = DATAØUT1(R)/2.0
      IF( BITCNT.LT.5 ) GØ TØ 51
      IF(ABS(DATAØUT1(R)).GT.0.1)QK(1)=SIGN(1.0,DATAØUT1(R))
      IF( ABS(DATAØUT1(R+2)).GT.0.1)QK(2)=SIGN(1.0,DATAØUT1(R+2))
      LEVELØUT = 0
      IF(SIGNLAST.EQ.QK(1).AND.SIGNLAST.EQ.QK(2)) LEVELØUT = 0
      IF(SIGNLAST.NE.QK(1).AND.SIGNLAST.NE.QK(2)) LEVELØUT = 1
      IF(SIGNLAST.EQ.QK(1).AND.SIGNLAST.NE.QK(2)) LEVELØUT = -1
      SIGNLAST = QK(2)
      LEVEL4(BITCNT-4) = LEVELØUT
      DA(5) = LEVELØUT/2.0
      IF(ABS(SIG(2)-DA(5) ) .GT. 0.1 ) DA(6) = 0.5 ; ERRØRCNT =
     * ERRØRCNT + 1
      IF(ABS(DATAØUT1(R)).GT.0.1)LEVEL1((BITCNT-5)*2+1) =
     * SIGN(1.0,DATAØUT1(R))
      IF(ABS(DATAØUT1(R+2)).GT.0.1) LEVEL1((BITCNT-5)*2+2) =
     * SIGN(1.0,DATAØUT1(R+2))
      DIFF1 = 0.0 ; DIFF2 = 0.0
      DIFF1 = ABS( DATAØUT1(R) - DATAØUT1(R-2))
      DIFF2 = ABS( DATAØUT1(R+2) - DATAØUT1(R) )
      IF( DIFF1 .LE. THRESHDF ) DIFF1 = 0.0
      IF( DIFF2 .LE. THRESHDF ) DIFF2 = 0.0
      IF( DIFFP1 .GT. THRESHDF ) GØ TØ 120
      IF( DIFF1 .GT. THRESHDF ) GØ TØ 121
  122 LEVEL2((BITCNT-5)*2+1) = LEVEL2P
      IF( DIFF2 .GT. THRESHDF ) GØ TØ 124
  123 LEVEL2((BITCNT-5)*2+2) = LEVEL2P
      DIFFP1 = 0.0
      GØ TØ 126
```

Figure 17B (Sheet 2 of 4). FORTRAN Listing of the Transmission, Receiving and Detection Process

```
  120 IF( DIFFP1 .GT. DIFF1 ) GØ TØ 125
      LEVEL2((BITCNT-5)*2) = LEVEL2P
  121 IF( DIFF1 .GT. ABS( DIFFP1) .AND. DIFFP1.LT.0.0 ) GØ TØ 129
      IF( DIFF1 .LT. DIFF2 ) GØ TØ 122
      LEVEL2((BITCNT-5)*2+1) = -LEVEL2P
      LEVEL2P = -LEVEL2P
      LEVEL2((BITCNT-5)*2+2) = LEVEL2P
      DIFFP1 = -DIFF2
      GØ TØ 126
  129 LEVEL2((BITCNT-5)*2-1) = -LEVEL2((BITCNT-5)*2-1)
      GØ TØ 122
  125 IF( DIFF2 .GT. DIFF1    .AND. DIFF2 .GT. THRESHDF ) GØ TØ 128
      LEVEL2((BITCNT-5)*2) =-LEVEL2P
      LEVEL2P = -LEVEL2P
      LEVEL2((BITCNT-5)*2+1) = LEVEL2P
      GØ TØ 123
  128 LEVEL2((BITCNT-5)*2) = LEVEL2P
      LEVEL2((BITCNT-5)*2+1) = -LEVEL2P
      LEVEL2P = -LEVEL2P
      LEVEL2((BITCNT-5)*2+2) = LEVEL2P
      DIFFP1 = 0.0
      GØ TØ 126
  124 DIFFP1 = DIFF2
  126 CØNTINUE
      IF( LEVEL2((BITCNT-5)*2+1).NE.ISIGN2) LEVEL2ØT = 1 ; GØ TØ 127
      IF( LEVEL2((BITCNT-5)*2+1).EQ.LEVEL2((BITCNT-5)*2+2))LEVEL2ØT=0 ;
     * GØ TØ 127
      LEVEL2ØT = -1
  127 ISIGN2 = LEVEL2((BITCNT-5)*2+2)
      IF( ABS( SIG(2) - LEVEL2ØT/2.0).GT.0.1) ERRCNT1 = ERRCNT1 + 1
      DIFF3 = DIFF4 = 0.0
      DIFF3 = ABS( DATAØUT1(R) + 0.5*DATAØUT1(R-2) - 0.5*DATAØUT1(R-4)
     *              - DATAØUT1(R-6) )
      DIFF4 = ABS( DATAØUT1(R+2) + 0.5*DATAØUT1(R) - 0.5*DATAØUT1(R-2)
     *              - DATAØUT(R-4) )
      IF( DIFFP .GT. THRESH2 ) GØ TØ 110
      IF( DIFF3 .GT. THRESH2 ) GØ TØ 111
  112 LEVEL3((BITCNT-5)*2+1) = LEVEL3P
      IF( DIFF4 .GT. THRESH2 ) GØ TØ 114
  113 LEVEL3((BITCNT-5)*2+2) = LEVEL3P
      DIFFP = 0.0
      GØ TØ 116
  110 IF( DIFFP .GT. DIFF3 ) GØ TØ 115
      LEVEL3((BITCNT-5)*2) = LEVEL3P
  111 IF( DIFF3 .LT. DIFF4 ) GØ TØ 112
      LEVEL3((BITCNT-5)*2+1) = -LEVEL3P
      LEVEL3P = -LEVEL3P
      GØ TØ 113
  115 LEVEL3((BITCNT-5)*2) = -LEVEL3P
      LEVEL3P = -LEVEL3P
      LEVEL3((BITCNT-5)*2+1) = LEVEL3P
      GØ TØ 113
  114 DIFFP = DIFF4
  116 CØNTINUE
      IF( BITCNT.LT.6 ) GØ TØ 51
      IF( LEVEL3((BITCNT-5)*2).NE.LEVEL3((BITCNT-5)*2-1))LEVEL3ØT = 1 ;
     1 GØ TØ 117
      IF( LEVEL3((BITCNT-5)*2).EQ.LEVEL3((BITCNT-5)*2+1))LEVEL3ØT = 0 ;
     1 GØ TØ 117
      LEVEL3ØT = -1
```

Figure 17B (Sheet 3 of 4). FORTRAN Listing of the Transmission, Receiving and Detection Process

```
      117 IF( ABS( SIG(1) - LEVEL3ØT/2.).GT.0.1) ERRCNT2 = ERRCNT2 + 1
          LEVEL6(BITCNT-5) = LEVEL3ØT
       51 SIG(1) = SIG(2)
          SIG(2) = SIG(3)
          SIG(3) = SIG(4)
          SIG(4) = SIG(5)
          SIG(5) = SIG(6)
          SIG(6) = DA(1)
X         CALL STRIPLØT(DA)
          DA(3) = DATAØUT(R+1)/2.0
          DA(4) = DATAØUT1(R+1)/2.0
X         CALL STRIPLØT(DA)
          DA(2) = P(2)/2.0
          DA(3) = DATAØUT(R+2)/2.0
          DA(4) = DATAØUT1(R+2)/2.0
X         CALL STRIPLØT(DA)
          DA(3) = DATAØUT(R+3)/2.0
          DA(4) = DATAØUT1(R+3)/2.0
X         CALL STRIPLØT(DA)
          BITCNT = BITCNT + 1
          DØ 11 I = 1,8
          DATAØUT(I) = DATAØUT(I+4)
       11 DATAØUT1(I) = DATAØUT1(I+4)
          DA(6) = 0.0
        9 CØNTINUE
          IF( N .LT. 2 ) GØ TØ 131
          WRITE(8) ( LEVEL4(L),L= 1,32 )
          WRITE(9) ( LEVEL6(L),L= 1,32 )
          DØ 130 KK = 1,32
          LEVEL4(KK) = LEVEL4(KK+32)
          LEVEL4(KK+32) = 0.0
          LEVEL6(KK) = LEVEL6(KK+32)
      130 LEVEL6(KK+32) = 0.0
          DØ 132 KK = 1,64
          LEVEL2(KK) = LEVEL2(KK+64)
          LEVEL2(KK+64) = 0.0
          LEVEL3(KK) = LEVEL3(KK+64)
      132 LEVEL3(KK+64) = 0.0
          BITCNT = BITCNT - 32
          WØRDCNT = WØRDCNT + 1
      131 WRITE(8) IDATA
          WRITE(9) IDATA
       10 CØNTINUE
      100 DØ 33 I = 1,6
       33 DA(I) = 0.0
X         CALL STRIPULSE(200)
X         CALL STRIPLØT(DA)
          PØWERAVG = PØWER/NP
          BITCNT = BITCNT + WØRDCNT*32
          ØUTPUT PØWERAVG, ERRØRCNT, ERRCNT1, BITCNT
          ØUTPUT ERRCNT2
          REWIND 7
          WRITE(8) ( LEVEL4(L), L = 1,32 )
          WRITE(9) ( LEVEL6(L), L = 1,32 )
          ENDFILE 8 ; REWIND 8
          ENDFILE 9 ; REWIND 9
          STØP
          END
```

Figure 17B (Sheet 4 of 4). FORTRAN Listing of the Transmission, Receiving and Detection Process

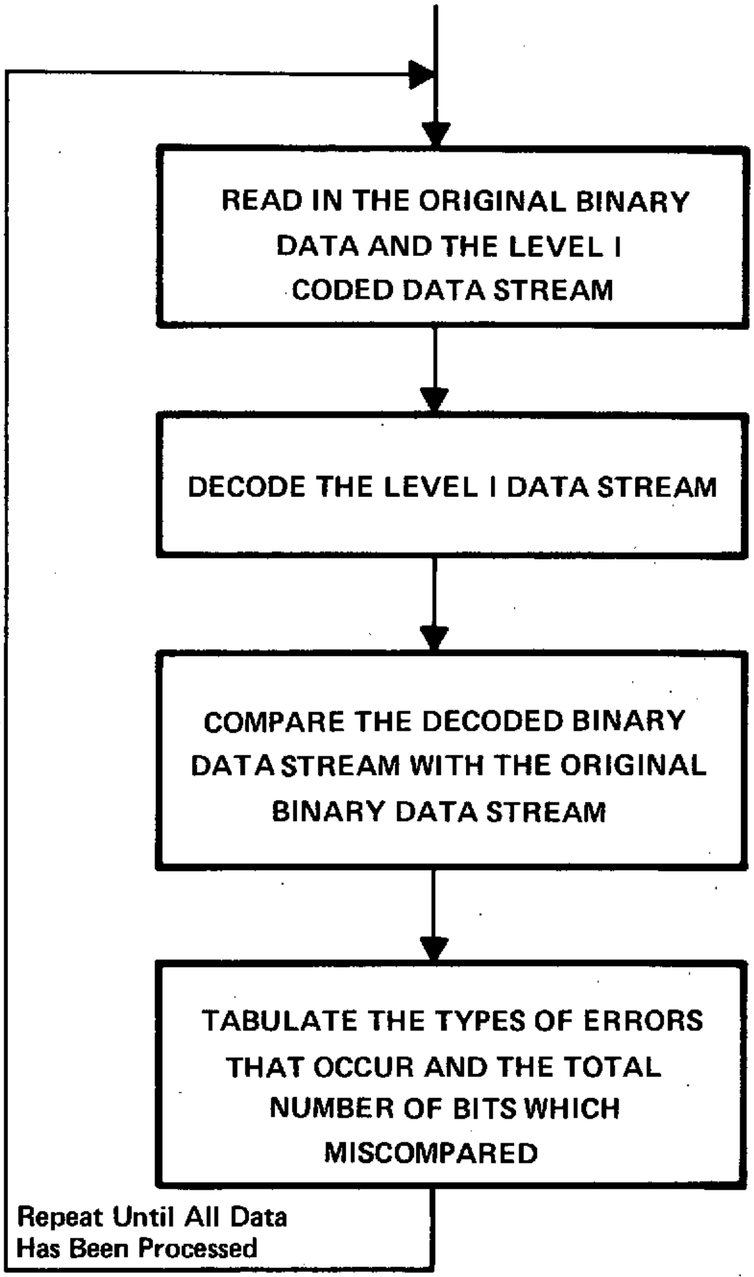


Figure 18A. Program to Decode Level I Data and Compare with Original Data

```
      INTEGER BITCNT
      DIMENSIØN LEVEL1IN(64), LEVELØUT(64)
      READ( 7,END=100 ) IDATA1 ; READ( 7 ) ( LEVEL1IN(J),J=1,32)
      READ( 7,END=100 ) IDATA2 ; READ( 7 ) ( LEVEL1IN(J),J=33,64)
      N = 1
      IDATAØUT = 0
   34 DØ 11 K = N,32
      IF( K.EQ.1 ) GØ TØ 32
   33 IF( LEVEL1IN(K-1).GT.-1 ) GØ TØ 30
      IF( LEVEL1IN(K).GT.0 ) WRITE( 6,300 )
      LEVELØUT(K) = 0 ; GØ TØ 40
  300 FØRMAT(T10,'A MID BIT TRANSITIØN NØT FØLLØWED BY A NØ TRANSITIØN')
   30 IF( LEVEL1IN(K-1).EQ.0 ) GØ TØ 31
      IF( LEVEL1IN(K).EQ.0 ) LEVELØUT(K) = 1 ; GØ TØ 40
      IF( LEVEL1IN(K).EQ.-1 ) LEVELØUT(K) = 0 ; GØ TØ 40
      WRITE( 6,301 ) ; LEVELØUT(K) = 0 ; GØ TØ 40
  301 FØRMAT( T10,'TWØ CØNSECUTIVE EDGE TRANSITIØNS DETECTED')
   31 IF( LEVEL1IN(K-1).NE.0 ) WRITE( 6,302 ) ; LEVELØUT(K) = 0 ;
     *     GØ TØ 40
   32 IF( LEVEL1IN(K).EQ.0 ) LEVELØUT(K) = 0 ; GØ TØ 40
      IF( LEVEL1IN(K).EQ.-1 ) LEVELØUT(K) = 1 ; GØ TØ 40
      IF( LEVEL1IN(K+1).EQ.0 ) LEVELØUT(K) = 1 ; GØ TØ 40
      IF( LEVEL1IN(K+1).EQ.-1 ) LEVELØUT(K) = 0 ; GØ TØ 40
      WRITE( 6,301 ) ; LEVELØUT(K) = 0 ; GØ TØ 40
  302 FØRMAT(T10,'AN ILLEGAL BIT TRANSITIØN CØDE DETECTED')
   40 IF( IFLAG.EQ.1 ) GØ TØ 52
      DATATEMP = ISC( LEVELØUT(K),32-K )
   11 IDATAØUT = IØR( IDATAØUT, DATATEMP )
      DATATEMP = IEØR( IDATA1,IDATAØUT )
      IF( DATATEMP.EQ.0 ) GØ TØ 50
      DØ 51 M = 1,32
      ERRØRCNT = IAND( DATATEMP,1 ) + ERRØRCNT
   51 DATATEMP = ISC( DATATEMP,-1 )
   50 BITCNT = BITCNT + 32
      K = 33 ; IFLAG= 1
      GØ TØ 33
   52 IFLAG = 0 ; N = 2
      DATATEMP = ISC( LEVELØUT(33), 31 )
      IDATAØUT = 0
      IDATAØUT = IØR( IDATAØUT, DATATEMP )
      DØ 53 J = 1,32
   53 LEVEL1IN(J) = LEVEL1IN(J+32)
      IDATA1 = IDATA2
      READ(7,END=101) IDATA2 ; READ(7,END=101)( LEVEL1IN(J),J=33,64)
      GØ TØ 34
  100 WRITE( 6,103 ) ; STØP
  103 FØRMAT(T10,'END ØF FILE ENCØUNTERED AT BEGINNING ØF TAPE ' )
  101 WRITE( 6,104 ) BITCNT , ERRØRCNT ; STØP
  104 FØRMAT( T10,I8,' BITS WERE PRØCESSED',I8,' ERRØRS WERE DETECTED')
      END
```

Figure 18B. FORTRAN Listing of the Decoding and Comparison Program

4.0 COMPARISONS AND DISCUSSION

4.1 Results for Manchester Bi-Phase

A Manchester bi-phase system may be defined and then optimized using the Nyquist-Sunde approach just as was done for Level I. In Section 4.1 the results (without any derivations) are given for a Manchester system with error detection capability. Such a system sacrifices some margin over noise in order to detect (without correcting) the majority of errors that occur. For this purpose, the signal train is sampled in the middle of each half-bit cell and compared against a zero threshold to determine the signal level in each half-bit independently. The intent is to have single half-bit errors show up as nonvalid Manchester pulses.

The optimum Nyquist-Sunde system is shown in Figure 19. The notation used is virtually identical with that of Figure 7. The Manchester encoding algorithm is simply that $s_k = +1$, $s_{k+1} = -1$ (k even) if the corresponding binary source bit was a logic 1, and $s_k = -1$, $s_{k+1} = +1$ if the binary source bit was a logic 0.

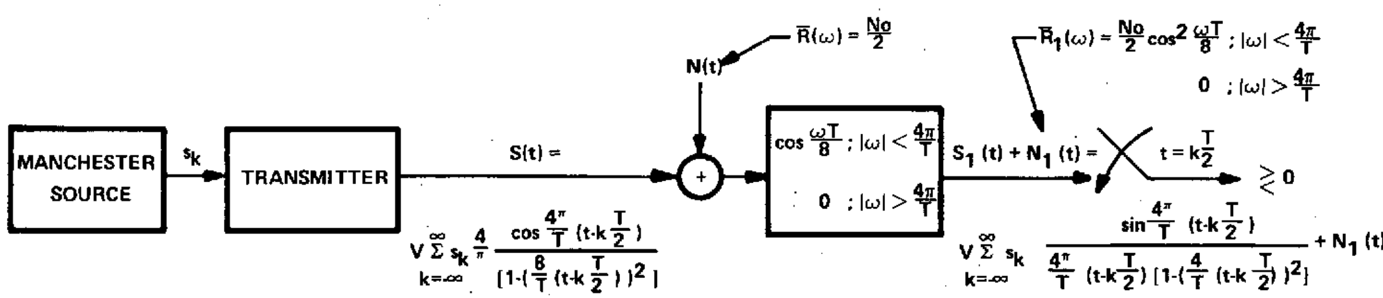


Figure 19. Manchester Bi-Phase System

The series for S(t) and $S_1(t)$ were summed in the following two special cases.

CASE 1 - All Manchester 0-Phases

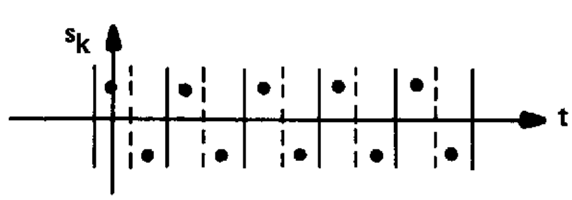


Figure 20. Case 1 Bit Stream

$$s_k = (-1)^k \ ; \ k = \ldots, -1, 0, 1, 2, \ldots \tag{76}$$

$$S(t) = \sqrt{2}\, V \cos \frac{2\pi t}{T} \tag{77}$$

$$S_1(t) = V \cos \frac{2\pi t}{T} \tag{78}$$

CASE 2 - Alternate Manchester 0- and π-Phases

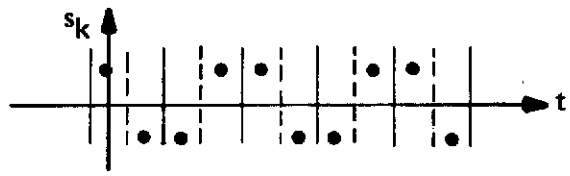


Figure 21. Case 2 Bit Stream

$$s_{2\ell-1} = s_{2\ell} = (-1)^\ell \ ; \ \ell = \ldots, -1, 0, 1, 2, \ldots \tag{79}$$

$$S(t) = \sqrt{2}\, V \left[\cos \pi/8 \, \cos\left(\frac{\pi t}{T} + \pi/4\right) + \cos \frac{3\pi}{8} \cos\left(\frac{3\pi t}{T} - \pi/4\right)\right] \tag{80}$$

$$S_1(t) = V \left[\left(\frac{\sqrt{2}+1}{2}\right) \cos\left(\frac{\pi t}{T} + \pi/4\right) + \left(\frac{\sqrt{2}-1}{2}\right) \cos\left(\frac{3\pi t}{T} - \pi/4\right)\right] \tag{81}$$

The average transmitted power in both Case 1 and Case 2 is

$$P_{AV} = V^2 \tag{82}$$

and the simulation confirms that this is the correct identification. The peak power required is that which occurs in Case 1 and is

$$P_{PK} = 2V^2 = 2\, P_{AV} \tag{83}$$

Finally, the bit error rate is

$$BER = 2 \cdot \frac{1}{2} \operatorname{erfc} \left(\sqrt{\frac{1}{2} \frac{P_{AV}}{N_o/T}}\right) = 2 \cdot \frac{1}{2} \left(\sqrt{\frac{1}{4} \frac{P_{PK}}{N_o/T}}\right) \tag{84}$$

4.2 Manchester Without Error Detection

If the error detection capability is relinquished, a more than 3 dB improvement in margin over noise may be effected for Manchester bi-phase. The simplest method of obtaining this improvement is to compare the difference between the first half-bit sample and the second half-bit sample against a zero threshold to determine whether the full bit was a zero phase or a pi-phase The effective signal power is thereby quadrupled while the noise power in the difference is only doubled. The effective signal-to-noise ratio is thus increased by a factor of two. The probability of error does not have to be doubled to convert from half bits to bits, and so the bit error rate is

$$\text{BER} = \frac{1}{2}\,\text{erfc}\left(\sqrt{\frac{P_{AV}}{N_o/T}}\right) = \frac{1}{2}\,\text{erfc}\left(\sqrt{\frac{1}{2}\,\frac{P_{PK}}{N_o/T}}\right) \tag{85}$$

Eq. (85) may be recognized as the bit error rate of an optimum performance communications system operating at bit rate $1/T$ in the presence of white Gaussian noise of spectral density N_0. The Manchester system of Section 4.2 is, therefore, an optimum system.

4.3 Other Level I Systems

The improvement obtained on using the difference of Manchester half-bit samples as a test statistic suggests that Level I systems employing sample differences might obtain an improvement over the system of Section 2. The simulation has been used to carry out a (nonexhaustive) study of such differencing schemes. The most successful of the schemes studied employed a linear combination of four neighboring half-bit samples to determine when the transitions between the levels take place. If the voltage output of the receiving filter, $S_1(t) + N_1(t)$, is called $V_1(t)$, then the sampled values are $V_1(k\,\frac{T}{2})$. (See Figure 7.) The statistic computed to determine whether a transition occurred between the k-1st and the kth half bit cell is

$$G_k \equiv \left| V_1\left((k+1)\,\frac{T}{2}\right) + \frac{1}{2}V_1\,(k\,\frac{T}{2}) - \frac{1}{2}\left((k-1)\,\frac{T}{2}\right) - V_1\left((k-2)\,\frac{T}{2}\right)\right| \tag{86}$$

As the Level I wave train undergoes a transition, G_k successively takes on the values 0, $\frac{1}{2}$ V, 7/4 V, 5/2V, 7/4V, 1/2 V, 0 (in the absence of noise). The 5/2 value is identified as the transition point. Identifying the transition point in the presence of noise is treated as a combined detection and estimation problem. G_k is compared against a threshold of 3/2 V in order to detect that a transition has taken place. To estimate the location of the transition points, the values of G_k above the threshold are examined to identify the local maxima. The results of the simulation are discussed below in Section 4.4.

Another Level I system that was studied for the sake of comparison was the system of Figure 19, with the Manchester source replaced by a Level I source. Since in this case, all the sample values are ± V (and $P_{AV} = V^2$), the bit error rate is (compare Eq. (72), (74), and (84)).

$$\mathrm{BER} = 2\alpha \cdot \frac{1}{2}\,\mathrm{erfc}\left(\sqrt{\frac{1}{2}\,\frac{P_{AV}}{N_o/T}}\right) \tag{87}$$

This system will be referred to as the "no intersymbol interference" (No I-I) Level I system in the discussion below.

Alpha is again the number of bits in error in the decoded Level I bit stream per half-bit error developed in the threshold comparison. The computation of α is now, however, more complex than in the arguments following Eq. (75). When a half-bit error occurs adjacent a transition, we may again expect approximately two bit errors to develop for each half-bit in error on the average. Now, however, since all the sample values are ±V, a significant number of half-bit errors also occur not adjacent to a transition. These errors generally result in nonvalid Level I wave trains and must be eliminated before (or during) Level I decoding. The Level I decoding algorithm may therefore be considered to correct many of the half-bit errors that occur not adjacent to a transition (the number corrected will be a function of the algorithm), and α will be significantly less than 2. Some crude estimates of α were made using the joint probabilities of Section 1 and a simple assumption about which errors are corrected. The results suggest that

$$\frac{1}{2} \lesssim \alpha \lesssim 1 \tag{88}$$

The simulation results for No I-I Level I presented below include a simple algorithm for obtaining a valid Level I wave train before decoding, and therefore include effects such as possible dependence of α on signal-to-noise ratio.

Still another Level I system was identified as worthy of study, but its analysis could not be completed within the scope of the current contract. It is shown in Figure 22.

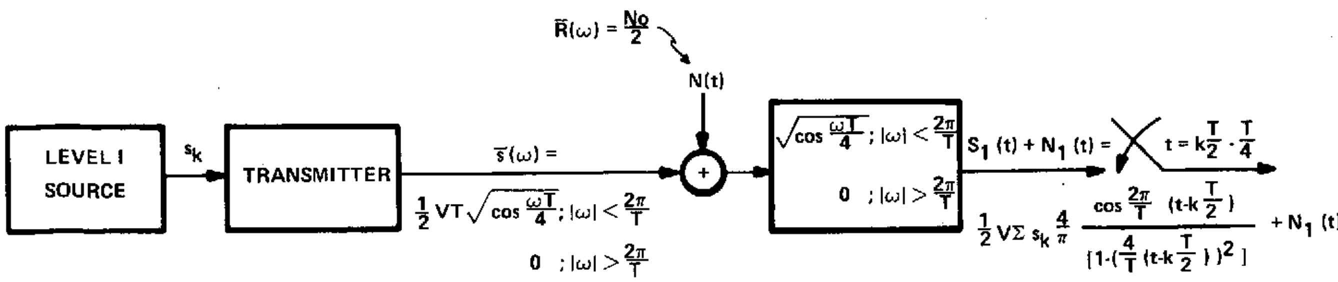


Figure 22. Level I, Transition Detection

The output pulse function

$$s_1(t) = \frac{4}{\pi}\cos 2\pi t/T \left[1 - \left(4t/T\right)^2\right]^{-1}$$

has the property that

$$s_1(\pm T/4) = 1, s_1(k\,T/2 - T/4) = 0 \quad \text{for } k \neq 0, 1.$$

Since the samples here are taken at the edges rather than at the centers of the half-bit cells, this system has the property that the sample values (in the absence of noise)

$$S_1\left(k\,\frac{T}{2} - \frac{T}{4}\right)$$

are $\pm V$ if there is no transition between the $k-1^{th}$ and the k^{th} half-bit cell and 0 if there is a transition. One way of detecting transitions, then, would be to compute the test statistic

$$\left|V_1\left(k\frac{T}{2} - T/4\right)\right|$$

where $V_1(t)$ is again $S_1(t) + N_1(t)$ and compare against a threshold of $1/2V$.

Finally, it may be noted that block and sequential decoding schemes have not been studied even though they might be expected to provide significant improvement in Level I performance because of the correlation present in the Level I bit stream. Such studies would, of course, be well outside the scope of the current contract. If or when they could be carried out, they would probably identify systems with better error performance at the cost of increased hardware complexity. Such systems would vitiate some of Level I's principal advantages; namely, simplicity and low cost.

4.4 Comparisons

The bit error rate of the Level I system studied in Section 2 is plotted in Figure 23, together with the BER's of the Section 4 systems (Manchester and Level I). The range 10^{-2} to 10^{-5} bit errors per bit is shown. The argument is taken to be $10 \log P_{AV}/N_o/T$, which is the ratio in dB of the average signal power to the noise power in a bandwidth equal to the information rate. The curves based on the analytic expressions derived are drawn solid while the curves drawn through points computed with the simulation are dashed. The simulation results were computed using, at most, 10,368 information bits, so that the lower points on the simulation curves have a greater variance.

The curve labeled "Manchester Without Error Detection" is a plot of Eq. (85). As mentioned in Section 4.2, it is an optimum performance curve. No system can obtain better performance (in white noise) without going to block or convolutional coding techniques.

The solid curve labeled "Manchester Bi-phase" is a plot of Eq. (84). The dashed curve was obtained with a Delco simulation. Although the development of this Manchester simulation was not part of the contract, the techniques employed were very similar to those of the Level I simulations. The curve is therefore shown to demonstrate the close agreement (better than 0.1 dB) between the simulation and the theory.

The solid curve labeled "Level I (Section 2)" is a plot of Eq. (74) using Eq. (72) with $\alpha = 2$. Recall that α is the average number of bit errors that result in the Level I decoded bit stream due to a half-bit error. (See the arguments following Eq. (75) in the text.) The simulation curve could be fit pretty closely by Eq. (74) with $\alpha = 1.4$. In fact, an exact fit could be obtained by an α that varied continuously from 1.2 at the upper end of the curve to 1.5 at the lower.

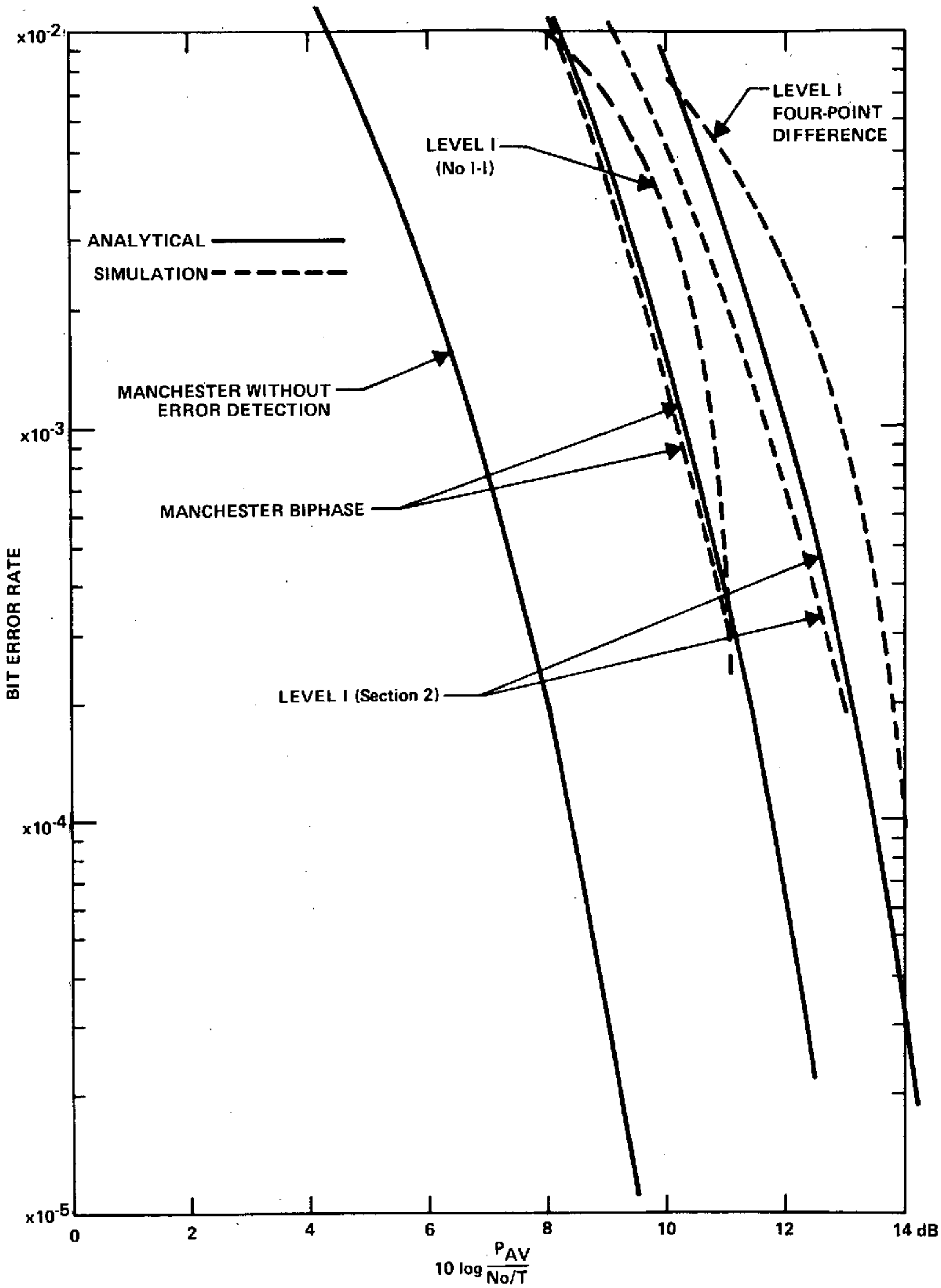


Figure 23. BER Comparisons

In comparing the curves, it is seen that a Level I, Section 2 system has between 1 and 1-1/2 dB less margin over noise that the Manchester bi-phase system of Section 4.1. To comment in detail and summarize the specific effects that lead to this result, refer to Eq. (84) for Manchester, namely

$$BER_{Man} = 2 \cdot \frac{1}{2} \mathrm{erfc}\left(\sqrt{\frac{1}{2} \frac{P_{AV}}{N_o/T}}\right) \tag{84}$$

and to Eq. (74) with the first term of Eq. (72) neglected.

$$BER_{Lev\,I} \simeq 2 \cdot \frac{3}{7}\, \alpha \cdot \frac{1}{2} \mathrm{erfc}\left(\sqrt{\frac{1}{3} \frac{P_{AV}}{N_o/T}}\right) \tag{74'}$$

The first term in Eq. (72) has been neglected because the ±V samples don't contribute significantly to the error rate over the range considered. The 3/7 factor is present because the ±V samples only occur 3/7 of the time. The factor α is discussed above and in the text following Eq. (75). Referring to the arguments of the square roots, there are three effects which result in Manchester having a 3/2 (or a 10 log 3/2 = 1.8 dB) advantage over Level I in signal-to-noise ratio.

- The Level I filter has half the Manchester filter bandwidth, so the Level I detection process is only contaminated by half the noise.
- The Level I samples are ±1/2V compared with ±V for Manchester; therefore, on a relative basis, Level I has 1/4 the signal power.
- Signal-to-noise ratio comparisons are made on the basis of average transmitted power. For Level I, $P_{AV} = 3/4V^2$ while for Manchester $P_{AV} = V^2$. Therefore, in converting to an average power basis for comparing signal-to-noise ratio, Manchester loses a factor 3/4 with respect to Level I

Putting these three effects together, Manchester is seen to have a $1/2 \times 4 \times 3/4 = 3/2$ advantage over Level I in signal-to-noise ratio.

The points on the curve labeled "Level I (four-point difference)" were computed with the simulation of the system described in Section 4.3. The four-point difference scheme is seen to have equal or poorer performance than the single-point Level I system of Section 2.

This result can be understood qualitatively as follows. The statistic of Eq. (86) has three times the noise variance of the single sample. In estimating the effective signal power, the detection and estimation part of the problem must be treated separately. In detecting a 5/2V value by comparing against a threshold of 3/2V, the effective signal power is V^2, an improvement over single-point Level I by a factor of 4. In estimating the location of the maxima, however, we are normally comparing 5/2V with 7/4V. One way to do this is to compute the difference and compare against a zero threshold. In computing the difference the noise variance is again doubled, while the effective signal power is now

$$(5/2V - 7/4V)^2 = 9/16\, V^2.$$

A crude estimate of the effective signal-to-noise ratio compared to single-point Level I is therefore

$$1/3 \times 4 \times 1/2 \times 9/16 = 3/8, \quad \text{or} \quad 10 \log 3/8 - 4.3 \text{ dB}.$$

The arguments are admittedly crude and, in fact, give a considerably poorer estimate of four-point differencing than is actually achieved on the simulation. On the basis of the simulation results, and to a lesser extent on these arguments, it is concluded that simple differencing schemes do not allow a significant improvement in Level I error performance.

The curve labeled "Level I (No I-I)" is drawn through the points generated by the simulation for the "no intersymbol interference" Level I system discussed in Section 4.3. Eq. (87) gives an analytical result of the bit error rate for this system. This equation was not plotted in Figure 23 because for $\alpha = 1$, it is identical with the Manchester bi-phase curve. Eq. (88), and the arguments preceding it suggest that $1/2 \leq \alpha \leq 1$, and the curves for other values of α can easily be inferred from the curve for $\alpha = 1$. Figure 23 shows two of the simulation points lying near the Manchester curve and the others lying above it. This suggests a rather strong dependence of α on signal-to-noise ratio. (Although the behavior of α, and therefore the Level I error correcting capability, depends on the particular decoding algorithm chosen, a study of decoding algorithms and attempts to optimize α were considered beyond the scope of the contract.) In any case, the Level I (No I-I) performance was, at most, 0.8 dB poorer than Manchester Bi-phase. It is felt that with proper algorithm design, this type of Level I system could be made to perform at least as well as Manchester over a specified range of BER. This is an improtant conclusion. Although the Level I system of Section 2 is optimum for its usual application in which it is desired to send information at the highest rate consistent with the bandwidth restriction, the No I-I Level I system of Section 4.3 obtains a 1 to 1-1/2 dB improvement in margin over noise by expanding the bandwidth by a factor of two. In applications in which this bandwidth expansion is allowed, the following statement may be made. The bit error rate performance of Level I is comparable to Manchester Bi-phase (with error detection) and only 3 dB poorer than Manchester without error detection.

APPENDIX C

EVALUATION OF DELCO ELECTRONICS 3-PHASE DPM AND CONVENTIONAL 4-PHASE DPM

1.0 INTRODUCTION

This appendix develops an expression for the phase error probability as a function of signal-to-noise ratio for L-phase differential phase modulation and presents an evaluation of Delco Electronics 3-phase DPM (differential phase modulation) technique versus conventional 4-phase DPM.

The results of the phase error probability analysis are shown in Figure C-1. As can be seen, at a signal-to-noise ratio of 10 dB, the probability of a phase error in 3-phase modulation is down by a factor of 10 from that of 4-phase. This factor increases with increasing SNR, and the advantages of 3-phase become more prominent. Translating phase error rate into bit error rate shows: (1) for conventional 4-phase, a single phase error produces, on the average, one bit error, while (2) for Delco's 3-phase, a phase error produces, on the average, approximately one and one-half bit errors. Comparing the two schemes at 10 dB, then, the 3-phase bit error rate is down by a factor of about seven from that of 4-phase.

2.0 DESCRIPTION OF THE MODULATION SCHEMES

A binary stream is used to modulate the phase of a carrier in a differential manner. Figure C-2 shows conventional 4-phase DPM, where 00 causes a phase increment of 45^o, 01 a phase increment of 135^o, etc. Figure C-3 shows Delco's 3-phase DPM, where two consecutive 1's cause a phase increment of 180^o, two consecutive 0's, a phase increment of 300^o, and 10 a phase increment of 60^o. The 01 combination is encoded by holding the previous phase for one bit time, thus coding the initial 0 bit, with the 1 being coded with the next bit in the data stream. An example of coding with conventional 4-phase and with Delco's 3-phase is shown in Figure C-4.

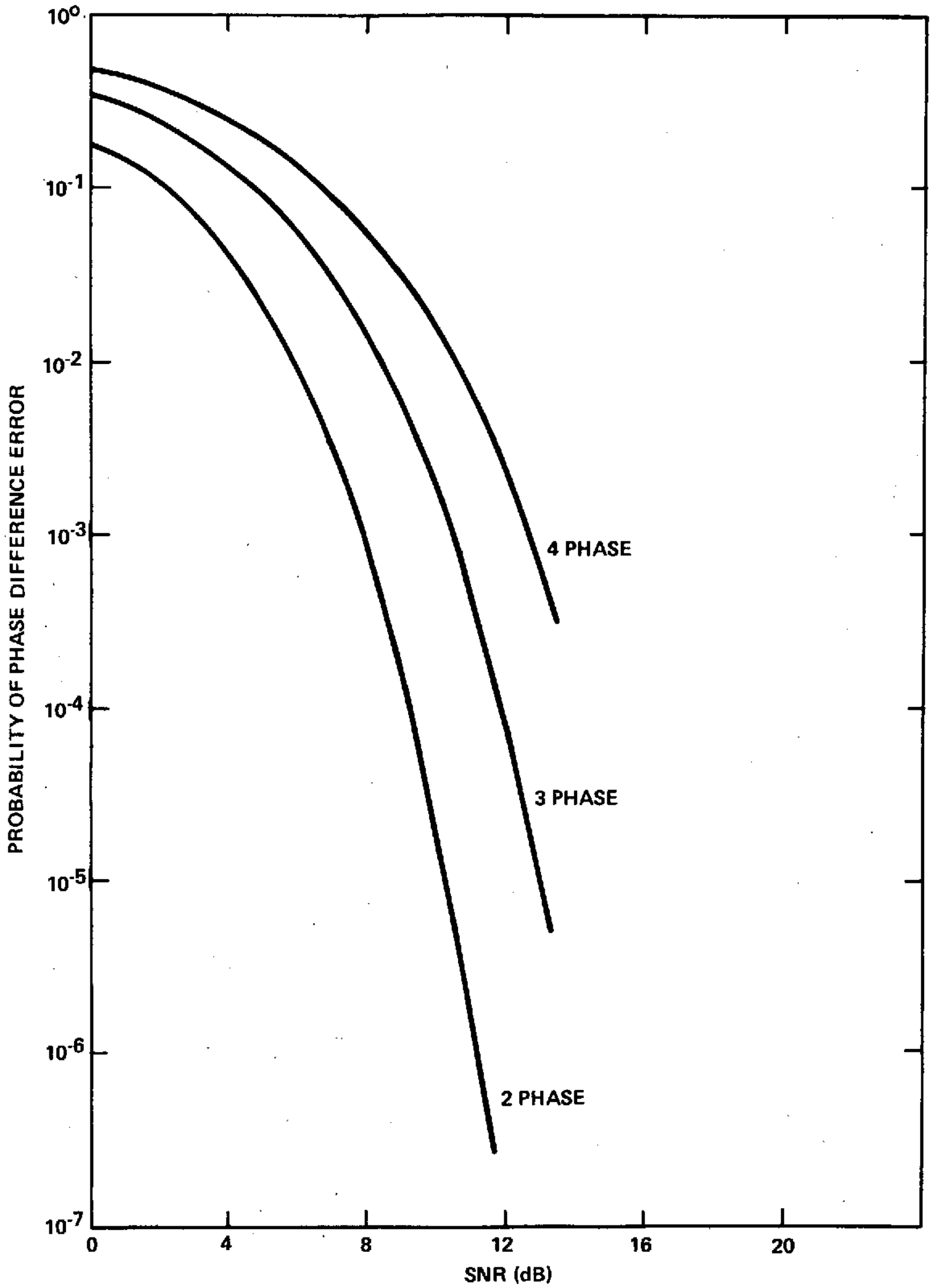


Figure C-1. Probability of Phase Difference Error

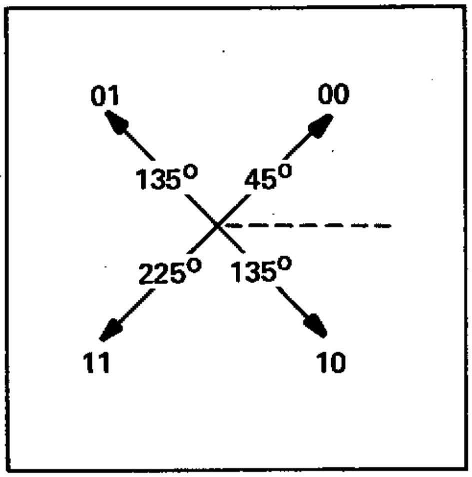


Figure C-2. Conventional 4-Phase DPM

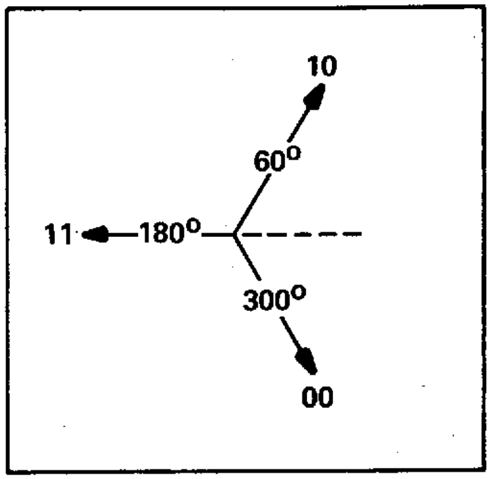


Figure 3. Delco's 3-Phase DPM

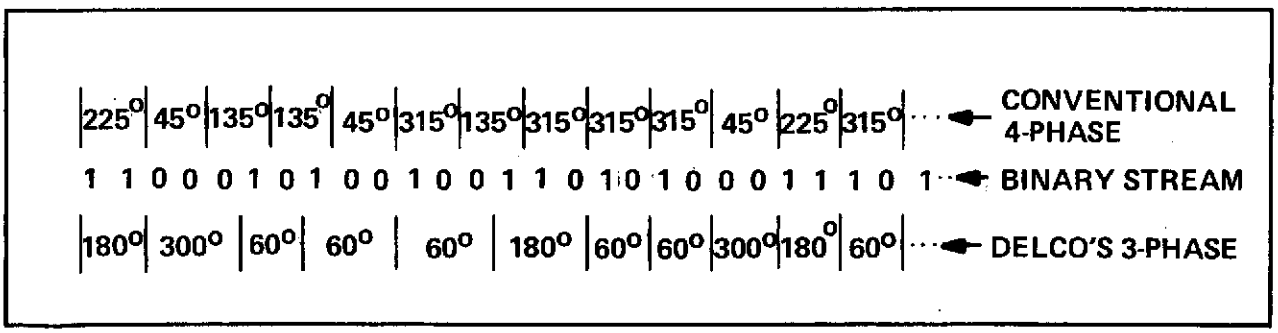


Figure C-4. An Example of Encoding in Conventional 4-Phase and in Delco's 3-Phase

3.0 PHASE ERROR PROBABILITY

This evaluation is made by first presenting a derivation of the probability of a phase difference error when the phase change between two signaling intervals is preserved as a reference (differential phase modulation). The traditional assumptions are made; namely, of an ideal digital phase detector observing a sine wave with one of L-phases in additive gaussian noise.

The probability density of the error in phase, θ, is given by:

$$p(\theta) = \frac{1}{2\pi} e^{-\rho} \left[1 + \sqrt{\pi\rho} \cos\theta \, e^{\rho \cos^2\theta} \left(1 + \text{erf}\,(\sqrt{\rho}\cos\theta)\right)\right] -\pi \le \theta \le \pi \qquad (1)$$

where ρ is the signal-to-noise ratio and erf denotes the error function. A derivation of Equation 1 is given in Section 5 of this appendix for completeness and because, even though it is reportedly a well known result, it is incorrect in the reference.

Errors are committed when the detector measures a phase outside of the region from $\phi - \pi/L$ to $\phi + \pi/L$, where ϕ is the received phase in the absence of noise.

The single error probability, P(e), in DPM is thus given by the probability that the absolute value of a phase difference exceeds π/L; that is,

$$P(e) = \text{Prob}\left(\pi/L < |\theta_1 - \theta_0|\right)$$

In order to derive an expression for P(e), it is assumed that the probability of θ lying between θ_1 and $\theta_1 + d\theta_1$, and of θ_o lying outside the interval from $\theta_1 - \pi/L$ to $\theta_1 + \pi/L$, is the product:

$$p(\theta_1)\, d\theta_1 \left[1 - \int_{\theta_1-\pi/L}^{\theta_1+\pi/L} p(\theta_o)\, d\theta_o\right]$$

Letting θ vary from $-\pi$ to $+\pi$ and recognizing the symmetry of $p(\theta)$ about $\theta=0$, yields

$$P(e) = 2\int_o^{\pi} p(\theta_1) \left[1 - \int_{\theta_o-\pi/L}^{\theta_o+\pi/L} p(\theta_o)\, d\theta_o\right] d\theta_1$$

Since phase differential is used to compute the transmitted phase in DPM, successive errors are not independent, that is

$$P(e_2 \mid e_1) \neq 0$$

for finite S/N. From the above, it follows that the joint probability of successive errors is given by

$$P(e_1, e_2) = 2 \int_0^{\pi} p(\theta_1) \left[1 - \int_{\theta_1 - \pi/L}^{\theta_1 + \pi/L} p(\theta_o)\, d\theta_o \right] \left[1 - \int_{\theta_1 - \pi/L}^{\theta_1 + \pi/L} p(\theta_2)\, d\theta_2 \right] d\theta_1$$

or in more compact notation

$$P(e_1, e_2) = 2 \int_0^{\pi} p(\phi) \left[1 - \int_{\phi - \pi/L}^{\phi + \pi/L} p(\alpha)\, d\alpha \right]^2 d\phi \qquad (2)$$

Recognizing that the conditional probability $P(e_2 \mid e_1)$ is given by the ratio of the joint to the single error probability, we have

$$P(e_2 \mid e_1) = \frac{P(e_1, e_2)}{P(e_1)}$$

and, thus, expressions for all pertinent probabilities.

We now show that the conditional error probability of Equation 2 reduces to the results of Salz and Saltzberg[2] for L = 2, namely,

$$P(e_1, e_2) = \frac{1}{2} \int_0^{\pi} \left[1 - \mathrm{erf}\, (\sqrt{\rho} \cos \theta) \right]^2 p(\theta)\, d\theta \qquad (3)$$

Letting

$$q(\phi) = \int_{\phi - \pi/2}^{\phi + \pi/2} p(\alpha)\, d\alpha$$

and differentiating with respect to ϕ

$$\frac{dq}{d\phi} = p(\phi + \pi/2) - p(\phi - \pi/2)$$

Since

$$\cos(\phi \pm \pi/2) = \mp \sin\phi$$

we can write

$$\frac{dq}{d\phi} = -\sqrt{\frac{\rho}{\pi}} \sin\phi \, e^{-\rho\cos^2\phi}$$

By making a change of variable

$$x = \sqrt{\rho}\cos\phi$$

and recognizing that $q(\pi/2) = 1/2$, we have

$$q(\phi) = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\sqrt{\rho}\cos\phi} e^{-x^2} dx$$

Since the integrand is an even function and

$$\frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} e^{-x^2} dx = 1$$

Therefore,

$$1 - q(\phi) = \frac{1}{\sqrt{\pi}} \int_{\sqrt{\rho}\cos\phi}^{\infty} e^{-x^2} dx = \frac{1}{2} [1 - \mathrm{erf}(\sqrt{\rho}\cos\phi)]$$

which proves the equality of Equation 2 and Equation 3 for $L = 2$.

Figure C-1 is a plot of the single error probability for $L = 2$, 3, and 4. The evaluation of the integral

$$P(e) = 2 \int_{o}^{\pi} p(\theta_1) \left[1 - \int_{\theta_1 - \pi/L}^{\theta_1 + \pi/L} p(\theta_o)\, d\theta_o \right] d\theta_1$$

requires numerical integration for the signal-to-noise ratios of interest. The double integration was carried out using Simpson's rule and an error function approximation accurate to nine digits.

4.0 BIT ERROR PROBABILITY

To evaluate the effectiveness of Delco Electronics 3-phase DPM versus conventional 4-phase DPM, the process by which phase errors are converted to bit errors must be examined. In conventional 4-phase (Figure C-2), considering nonadjacent phase vector errors as highly improbable, phase errors translate to bit errors one-to-one. Three-phase modulation as shown in Figure C-3 results in a higher bit-to-phase error rate. Considering first the no-hold combinations, a "11" could be interpreted as either a "10" or "00" with equal probability, which results in an average of one and one-half bit errors per phase error. The same is true for a 00 interpreted as a 11 or 10. A 10 interpreted as a 11 or 00 results in one bit error per phase error. When a 01 follows a 11 or 10, the phase angle is held for one additional bit time. To investigate the bit-to-phase error rate for this case, the six combinations of five binary digits beginning 1101, 0001, or 1001 must be analyzed. There are 18 ways that these holding operations can result in bit errors. Considering these, it can be determined that the 18 possible phase errors produce 33 bit errors. Holding operations for random data occur with a relative frequency of one-fourth. The bit-to-phase error ratio for Delco's 3-phase is thus

$$\frac{1}{4}\left(\frac{33}{88}\right) + \frac{3}{4}\left(\frac{8}{6}\right) = 1.46$$

Thus the probability of bit error for Delco's 3-phase is a factor of about one and one-half higher than the probability of phase error shown in Figure C-1. For biphase and 4-phase, the probabilities of phase and bit error are equal. Comparing 3-phase and 4-phase at 10 dB then, the 3-phase bit error rate is down by a factor of about seven from that of 4-phase.

5.0 DERIVATION OF PROBABILITY DENSITY OF PHASE IN PHASE MODULATION

The received wave is assumed to have the form

$$V(t) = (A + x)\cos(\omega t + \phi) + y \sin(\omega t + \phi)$$

where ϕ is the phase that would be measured by the ideal phase detector at the end of a t-second signalling interval in the absence of noise, and $x = x(t)$, and $y = y(t)$ are the inphase and quadrature Gaussian noise components, respectively, each with average noise power of σ^2. The phase of the received wave is then

$$\text{Phase of } V(t) = \phi + \tan^{-1}\frac{y}{A + x}$$

The next step is to find the probability density function (pdf) on the error in phase, $p(\theta)$, where

$$\theta = \tan^{-1} \frac{y}{A+x}$$

Letting

$$u = A + x$$

$$v = y$$

the pdf on u and on v are given by

$$f(u) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-(u-A)^2/2\sigma^2}$$

and

$$g(v) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-v^2/2\sigma^2}$$

Recognizing that $\theta = \tan^{-1} v/u$, one can transform to polar coordinates, ($u = r\cos\theta$ and $v = r\sin\theta$) to find the joint density function in r and θ, $q(r,\theta)$, with $p(\theta)$ then being given by the integral

$$p(\theta) = \int_0^\infty q(r,\theta)\, dr$$

Following this procedure, we obtain an expression for $q(r,\theta)$

$$q(r,\theta) = \frac{1}{2\pi\sigma^2} e^{-A^2/2\sigma^2} \left[re^{-(r^2-2rA\cos\theta)/2\sigma^2} \right]$$

Letting the signal-to-noise ratio be

$$\rho = A^2/2\sigma^2$$

and

$$s^2 = \frac{1}{2\sigma^2} (r^2 - 2rA\cos\theta + A^2\cos^2\theta)$$

the pdf on phase error is given by

$$p(\theta) = \frac{1}{\sqrt{2}\,\pi\sigma} e^{-\rho(1-\cos^2\theta)} \int_{-\sqrt{\rho}\cos\theta}^{\infty} (\sqrt{2}\sigma s + A\cos\theta)\, e^{-s^2}\, ds$$

which becomes an integration

$$p(\theta) = \frac{1}{2\pi} e^{-\rho} \left[1 + \sqrt{\pi\rho} \cos \theta^{\rho \cos^2\theta} (1 + \mathrm{erf}(\sqrt{\rho}\cos\theta))\right]$$

where erf denotes the error function

$$\mathrm{erf}(\alpha) = \frac{2}{\sqrt{\pi}} \int_0^{\alpha} e^{-x^2} dx.$$

6.0 REFERENCES

1. Lucky, Salz and Weldon, "Principles of Data Communications," McGraw-Hill 1968, Chapter 9.

2. Salz and Saltzbert, "Double Error Rates in Differentially Coherent Phase Systems," IEEE Transactions on Communication Systems, June 1964, Vol. CS-12, No. 2, pp 202-205.

APPENDIX D

CONVOLUTIONAL ENCODING AND VITERBI DECODING

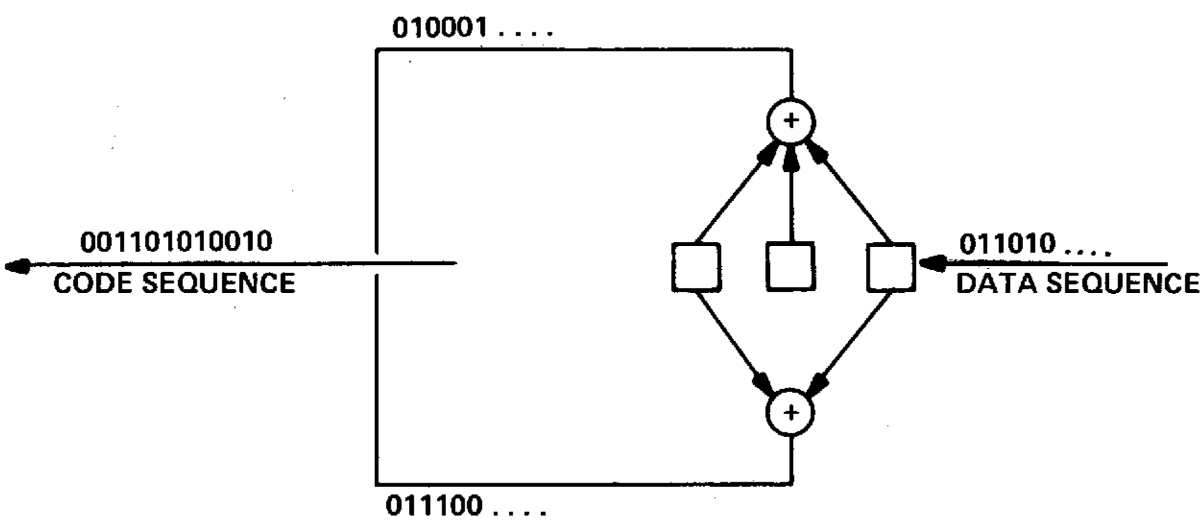
1.0 INTRODUCTION

This appendix presents: (1) the essential features of convolutional encoding and Viterbi decoding, (2) a listing of a digital computer program used to evaluate the compatibility of Level I encoding and decoding with the above mentioned techniques, and (3) the results of simulations with random data sequences.

2.0 CONVOLUTIONAL ENCODER

A general binary convolutional encoder consists of a bK stage binary shift register and v mod-2 adders. Each of the mod-2 adders is connected to certain of the shift register stages. The pattern of connections specifies the code. Information bits are shifted into the encoder shift register b bits at a time. After each b bit shift, the outputs of the mod-2 adders are sampled sequentially yielding the code symbols.

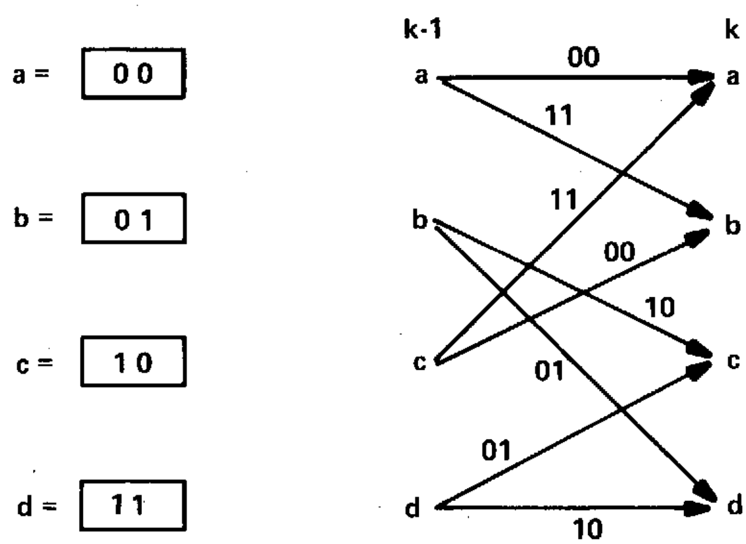
An example of a simple convolutional encoder with K = 3, b = 1, and v = 2 is shown below. The binary input data bits 0, 1, 1, 0, 1, 0... generate the code outputs 00, 11, 01, 00, 10, ...



The state of the convolutional encoder is the contents of the first b(k-1) shift register stages. The encoder state together with the next b input bits uniquely specify the v output symbols.

In the above example, the encoder state is specified by the contents of the first two shift register stages. Denoting the four possible states as a = 00, b = 01, c = 10, and d = 11, the input bits 0, 1, 1, 0, 1, 0... then correspond to the successive encoder states a, b, d, c, b, c...

The allowable transitions between states at k - 1 and k are shown in the diagram below together with the transition code sequence. In this example, two paths lead to each of the four states. In general, there are 2^b paths entering each of the $2^{b(K-1)}$ states.



Assuming a binary symmetric channel (i.e., symbol errors are independent and occur with probability (p) and that all input data sequences are equally likely, a maximum likelihood decoder is one which examines the error-corrupted received sequence and chooses the data sequence corresponding to the transmitted code sequence that is closest to the received sequence in the sense of Hamming distance; that is, the transmitted sequence which differs from the received sequence in the minimum number of symbols.

A brute-force maximum likelihood decoder would calculate the Hamming distance on all paths and select the minimum. The information bits corresponding to that path would form the decoder output. For an L bit information sequence there are 2^L paths in the above example. This method obviously quickly becomes impractical as L increases.

3.0 VITERBI DECODING

The Viterbi decoding algorithm greatly reduces the effort required for maximum likelihood decoding by taking advantage of the fact that the minimum distance path to, for example, state a at time k in the above encoder example can be only one of two candidates: (1) the minimum distance path to state a at time k-1, and (2) the minimum distance path to state c at time k-1. A comparison is performed by adding the new distance accumulated in the k^{th} transition by each of these paths to this minimum distances at time k-1.

In general, a Viterbi decoder calculates the likelihood of each of the 2^b paths entering a given state and eliminates from further consideration all but the most likely path that leads to that state. This is done for each of the $2^{b(K-1)}$ states; after each decoding operation, only one path remains leading to each state. In cases where distances are identical, an arbitrary selection is made. It is important to recognize that in eliminating the less likely paths entering each state, the Viterbi decoder will not reject any path that would have been selected by the brute-force maximum likelihood decoder.

The great advantage of the Viterbi decoder is that the number of decoder operations performed in decoding L bits in only $L2^{b(K-1)}$, which is linear in L. The Viterbi decoding technique is limited to relatively short constraint length codes ($K \leq 6$) due to the exponential dependence of decoder operations per bit decoded on K.

It has been shown that, with high probability, the $2^{b(k-1)}$ decoder-selected paths have a common stem, which eventually branches off to the various states. This further suggests that, if the decoder stores enough of the past information bit history of each of the $2^{b(K-1)}$ paths, the oldest bits on all paths will be identical. It has been demonstrated theoretically and through simulation that a value of the length of the information bit path history per state of four or five times the code constraint length, K, is sufficient for negligible degradation from optimum decoder performance.

4.0 SIMULATION PROGRAM

A digital simulation program was developed incorporating rate 1/2 convolutional and Level 1 encoding of random source data, error injection capability, Level 1 and Viterbi decoding. A listing of this program follows at the end of this appendix.

5.0 COMPATABILITY OF CONVOLUTIONAL AND LEVEL I ENCODING

A number of computer simulations have been made in an attempt to determine the compatibility of convolutional and Level I encoding. Two programs were employed. The first permits rate one-half convolutional encoding of random source data, error injection, and Viterbi decoding. The second, discussed above, incorporates rate one-half convolutional encoding of random source data followed by Level I encoding, error injection, Level I decoding, and Viterbi decoding. The Level I decode uses the essential features of the Viterbi algorithm. Bit memory paths of length 16 and 32 bits were used in the Level I and Viterbi decoding, respectively.

Each run was made with a source data sequence of 4,500 random bits. Error rates were established which were high enough to result in errors in the decoded sequence. Based on the models of communication links discussed in Section 4 of Appendix B, it was concluded that the probability of a channel error in the case of rate onehalf convolutional transmission is given by

$$\Pr(e) = \frac{1}{2}\,\mathrm{erfc}\sqrt{\frac{P_{AV}}{2N_0/T}}$$

whereas in the case of rate one-half convolutional plus Level I transmission, the probability is determined by

$$\Pr(e) = \frac{1}{2}\,\mathrm{erfc}\sqrt{\frac{P_{AV}}{4\,N_0/T}}$$

From these equations, a signal-to-noise ratio which results in 700 errors in 4,500 bits of source data for rate one-half convolutional transmission while results in 1,420 errors in rate one-half convolutional plus Level I transmission.

The results of simulations to test the compatibility of the two encoding schemes is shown in Table 1. For the lower error rates used, it can be concluded that the convolutional encoding and Viterbi decoding is more effective in error removal. At the higher error rates, the results tend to indicate that the inclusion of Level I encode-decode does not affect the error correcting capability of Viterbi decoding.

RATE 1/2 CONVOLUTIONAL ENCODER VITERBI DECODER		RATE 1/2 CONVOLUTIONAL AND LEVEL I ENCODER LEVEL I AND VITERBI DECODER	
CHANNEL ERRORS	ERRORS AFTER DECODE	CHANNEL ERRORS	ERRORS AFTER DECODE
700	73	1,420	308
800	129	1,540	400
900	194	1,640	439
1,000	204	1,760	459
1,100	382	1,840	512
1,200	421	1,940	532
1,300	516	2,040	564
1,400	585	2,120	560

Table 1. Results of Simulations for Compatibility Test

6.0 REFERENCES

1. J.A. Heller and I.M. Jacobs, "Viterbi Decoding for Satellite and Space Communication," IEEE Trans. Commun. Tech., Vol. COM-19, Oct 1971, pp. 835-848
2. A.J. Viterbi, "Convolutional Codes and Their Performance in Communication Systems," IEEE Trans. Commun. Tech., Vol. COM-19, Oct 1971, pp. 751-772.

SIMULATION PROGRAM

```
C           THIS PROGRAM IS FOR THE CONVOLUTIONAL ENCODING OF
C           BINARY DIGITS LISTED IN THE MATRIX R, WHERE K=3,
C           B=1, V=2 ARE THE CONVOLUTIONAL PARAMETERS
C
       IMPLICIT INTEGER(A-Z)
       COMMON  A1(640),A2(640),PS(640),DR(640),RO(640)
       DIMENSION RAW(400)
       DIMENSION R(160)
       INTEGER R
       READ(5,100) (R(I),I=1,160)
   100 FORMAT(80I1)
       CALL CENCOD(R)
C
C  PACK  A1,A2  INTO RAW
                               DO 5  I=1,320,2
       RAW(I)=     A1(I/2+1)
     5 RAW(I+1)=   A2(I/2+1)
       CALL ENCODE (320,RAW)
C  ERROR INTRODUCTION
C                          ERROR INTRODUCTION
       INTEGER ERROR
       IX=67
       WRITE(6,172)
   172 FORMAT(1H1)
       DO 1002  I=1,13
       CALL RANDOM(IX,YFL)
       ERROR =MOD(IABS(IX),250)
       WRITE(6,1003)  ERROR,IX
  1003 FORMAT(' A1  A2  ERROR AT' ,2I4)
       IF(MOD(IX,2).EQ.0)          A1(ERROR      )=MOD(A1(ERROR)+1,2)
       IF(MOD(IX,2).EQ.1)          A2(ERROR      )=MOD(A2(ERROR)+1,2)
  1002  CONTINUE
       CALL DECODE (320,RAW)
C  SPREAD RAW BACK TO A1,A2
       DO  6  I=1,320,2
       A1(I/2+1 )   =       RO (I)
     6 A2(I/2+1 )   =       RO (I+1)
       CALL  VDECOD(R)
       END
```

```
      SUBROUTINE CENCOD(R)
      IMPLICIT INTEGER(A-Z)
      COMMON  A1(640),A2(640),PS(640),DR(640),RO(640)
      DIMENSION  R(160)
      INTEGER A1
      INTEGER A2
C          THE SEQUENCE A1(I)A2(I) REPRESENTS THE CODE FOR R(I)
C
      A1(1) = R(1)
      A2(1) = R(1)
      A1(2) = R(1) + R(2)
      A2(2) = R(2)
      IF(A1(2).EQ.2) A1(2) = 0
      DO 200 I=1,158
      A1(I+2) = R(I) + R(I+1) + R(I+2)
      A2(I+2) = R(I) + R(I+2)
      IF(A1(I).EQ.2) A1(I) = 0
      IF(A1(I).EQ.3) A1(I) = 1
      IF(A2(I).EQ.2) A2(I) = 0
  200 CONTINUE
      DO 300 I=1,160
      PRINT 400,I,R(I),A1(I),A2(I)
  400 FORMAT(I10,I5,I5,I1)
  300 CONTINUE
      RETURN
      END
```

VITERBI DECODING

```
      SUBRØUTINE VDECØD(R)
      IMPLICIT INTEGER(A-Z)
      CØMMØN  A1(640),A2(640),PS(640),DR(640),RØ(640)
      DIMENSIØN R(1)
C          THIS PRØGRAM IS FØR THE VITERBI DECØDING ØF A
C          CØNVØLUTIØNAL CØDE ØF BINARY DIGITS, WHERE K=3,
C          B=1, V=2 ARE THE CØDE PARAMETERS
C
      DIMENSIØN BM(4,160),HD(4,160),RD(160),DBM(4,160)
      INTEGER BM
      INTEGER DBM
      INTEGER HD
      INTEGER RØ
      INTEGER RD
C
C          N IS THE DESIRED LENGTH ØF THE BIT MEMØRY PATH PER PATH
C
      READ (5,600) N
  600 FØRMAT(I3)
C
C          INITIALIZATIØN ØF DECØDING
C
      HD(1,1) = HD(2,1) = HD(3,1) = HD(4,1) = 0
      HD(1,2) = HD(2,2) = HD(3,2) = HD(4,2) = 0
      SUM1 = A1(1) + 1
      IF(SUM1.EQ.2) SUM1 = 0
      SUM2 = A2(1) + 1
      IF(SUM2.EQ.2) SUM2 = 0
      SUM3 = A1(2) + 1
      IF(SUM3.EQ.2) SUM3 = 0
      SUM4 = A2(2) + 1
      IF(SUM4.EQ.2) SUM4 = 0
      SUM5 = A1(3) + 1
      IF(SUM5.EQ.2) SUM5 = 0
      SUM6 = A2(3) + 1
      IF(SUM6.EQ.2) SUM6 = 0
      HD1 = A1(1) + A2(1) + A1(2) + A2(2) + A1(3) + A2(3)
      HD2 = SUM1 + SUM2 + SUM3 + A2(2) + SUM5 + SUM6
      HD3 = A1(1) + A2(1) + A1(2) + A2(2) + SUM5 + SUM6
      HD4 = SUM1 + SUM2 + SUM3 + A2(2) + A1(3) + A2(3)
      HD5 = A1(1) + A2(1) + SUM3 + SUM4 + SUM5 + A2(3)
      HD6 = SUM1 + SUM2 + A1(2) + SUM4 + A1(3) + SUM6
      HD7 = A1(1) + A2(1) + SUM3 + SUM4 + A1(3) + SUM6
      HD8 = SUM1 + SUM2 + A1(2) + SUM4 + SUM5 + A2(3)
C     PRINT 650,HD1,HD2,HD3,HD4,HD5,HD6,HD7,HD8
  650 FØRMAT(8I5)
      HD(1,3) = HD2
C
C          BM(1,I) THRU BM(4,I) REPRESENT THE BIT MEMØRY PATHS
C          AND HD(1,I) THRU HD(4,I) REPRESENT THE CØRRESPØNDING
C          HAMMING DISTANCE PATHS
C
      BM(1,1) = 1
```

```
           BM(1,2) = 0
           BM(1,3) = 0
           IF(HD1.LE.HD2) HD(1,3)=HD1;BM(1,1) = 0;BM(1,2) = 0;BM(1,3) = 0
           HD(2,3) = HD4
           BM(2,1) = 1
           BM(2,2) = 0
           BM(2,3) = 1
           IF(HD3.LE.HD4) HD(2,3)=HD3; BM(2,1) =0; BM(2,2) = 0;BM(2,3) = 1
           HD(3,3) = HD6
           BM(3,1) = 1
           BM(3,2) = 1
           BM(3,3) = 0
           IF(HD5.LE.HD6) HD(3,3)=HD5;BM(3,1) = 0;BM(3,2) = 1;BM(3,3) = 0
           HD(4,3) = HD8
           BM(4,1) = 1
           BM(4,2) = 1
           BM(4,3) = 1
           IF(HD7.LE.HD8) HD(4,3)=HD7;BM(4,1) = 0;BM(4,2) = 1;BM(4,3) = 1
     C     PRINT 660,HD(1,3),HD(2,3),HD(3,3),HD(4,3)
      660  FORMAT(4I5)
     C
     C          *    *    *    *    *    *    *    *    *    *    *    *
     C
           DO 10 I=4,160
           L = I - 1
           SUM1 = A1(I) + 1
           IF(SUM1.EQ.2) SUM1 = 0
           SUM2 = A2(I) + 1
           IF(SUM2.EQ.2) SUM2 = 0
           HD1 = HD(1,L) + A1(I) + A2(I)
           HD2 = HD(3,L) + SUM1 + SUM2
           HD3 = HD(1,L) + SUM1 + SUM2
           HD4 = HD(3,L) + A1(I) + A2(I)
           HD5 = HD(2,L) + SUM1 + A2(I)
           HD6 = HD(4,L) + A1(I) + SUM2
           HD7 = HD(2,L) + A1(I) + SUM2
           HD8 = HD(4,L) + SUM1 + A2(I)
     C     PRINT 670,I,HD1,HD2,HD3,HD4,HD5,HD6,HD7,HD8
      670  FORMAT(9I5)
     C
     C          DBM(I,J) IS A DUMMY BIT MEMORY PATH MATRIX
     C
           BM(1,I) = 0
           BM(2,I) = 1
           BM(3,I) = 0
           BM(4,I) = 1
           M1 = 1
           M2 = I
           IF(I.GT.N) M1 = I - N + 1
           DO 20 M=M1,M2
           DBM(1,M) = BM(1,M)
```

```
          DBM(2,M) = BM(2,M)
          DBM(3,M) = BM(3,M)
          DBM(4,M) = BM(4,M)
    20    CONTINUE
          M2 = I-1
          IF(HD1.LE.HD2) GO TO 672
          HD(1,I) = HD2
          DO 671 J=M1,M2
          BM(1,J) = DBM(3,J)
   671    CONTINUE
          GO TO 673
   672    HD(1,I) = HD1
   673    IF(HD3.LE.HD4) GO TO 675
          HD(2,I) = HD4
          DO 674 J=M1,M2
          BM(2,J) = DBM(3,J)
   674    CONTINUE
          GO TO 677
   675    HD(2,I) = HD3
          DO 676 J=M1,M2
          BM(2,J) = DBM(1,J)
   676    CONTINUE
   677    IF(HD5.LE.HD6) GO TO 679
          HD(3,I) = HD6
          DO 678 J=M1,M2
          BM(3,J) = DBM(4,J)
   678    CONTINUE
          GO TO 682
   679    HD(3,I) = HD5
          DO 681 J=M1,M2
          BM(3,J) = DBM(2,J)
   681    CONTINUE
   682    IF(HD7.LE.HD8) GO TO 684
          HD(4,I) = HD8
   683    CONTINUE
          GO TO 685
   684    HD(4,I) = HD7
          DO 685 J=M1,M2
          BM(4,J) = DBM(2,J)
   685    CONTINUE
          IF(I.LT.N) GO TO 10
C
C              DETERMINES OLDEST BIT ON THE MOST LIKELY PATH
C
          J = I
          K = I - N + 1
          IF(HD(1,J).LE.HD(2,J)) GO TO 740
   760    IF(HD(3,J).LE.HD(4,J)) GO TO 765
          IF(HD(2,J).LE.HD(4,J)) GO TO 710
          GO TO 730
   765    IF(HD(2,J).LE.HD(3,J)) GO TO 710
```

```
      GØ TØ 720
  740 IF(HD(3,J).LE.HD(4,J)) GØ TØ 770
      IF(HD(1,J).LE.HD(4,J)) GØ TØ 700
      GØ TØ 730
  770 IF(HD(1,J).LE.HD(3,J)) GØ TØ 700
      GØ TØ 720
C
C          RØ(I) IS DECØDE
C
  700 RØ(K) = BM(1,K)
      GØ TØ 800
  710 RØ(K) = BM(2,K)
      GØ TØ 800
  720 RØ(K) = BM(3,K)
      GØ TØ 800
  730 RØ(K) = BM(4,K)
  800 CØNTINUE
 10   CØNTINUE
C     DØ 696 I=1,160
C     PRINT 695,I,BM(1,I),BM(2,I),BM(3,I),BM(4,I)
C     PRINT 695,I,HD(1,I),HD(2,I),HD(3,I),HD(4,I)
  695 FØRMAT(5I5)
  696 CØNTINUE
      DØ 950 I=1,160
      RD(I) = RØ(I) - R(I)
      PRINT 900,I,RØ(I),R(I),RD(I)
  900 FØRMAT(I10,3I5)
  950 CØNTINUE
      RETURN
      END
```

LEVEL I ENCODE

```
      SUBROUTINE ENCODE(NBITS,R   )
      IMPLICIT INTEGER (A-Z)
      DIMENSION R(1)
      COMMON  A1(640),A2(640),PS(640),DR(640),RO(640)
      DR(1)=DR(2)=DR(3)=0
      DO 300 I=4,NBITS+3
      DR(I)=R(I-3)
300   CONTINUE
      DO 400 I=1,NBITS+3
      R(I)=DR(I)
400   CONTINUE
      PS(1)=1
      DO 1 I=1,NBITS
      J=I+1
      K=I+2
      IF(R(I).EQ.0.AND.R(J).EQ.0.AND.R(K).EQ.0.AND.PS(I).EQ.6) GO TO 2
      IF(R(I).EQ.0.AND.R(J).EQ.0.AND.R(K).EQ.0.AND.PS(I).EQ.3) GO TO 2
      IF(R(I).EQ.0.AND.R(J).EQ.0.AND.R(K).EQ.0.AND.PS(I).EQ.1) GO TO 3
      IF(R(I).EQ.0.AND.R(J).EQ.0.AND.R(K).EQ.0.AND.PS(I).EQ.7) GO TO 3
      IF(R(I).EQ.0.AND.R(J).EQ.0.AND.R(K).EQ.0.AND.PS(I).EQ.2) GO TO 4
      IF(R(I).EQ.0.AND.R(J).EQ.0.AND.R(K).EQ.1.AND.PS(I).EQ.6) GO TO 5
      IF(R(I).EQ.0.AND.R(J).EQ.0.AND.R(K).EQ.1.AND.PS(I).EQ.1) GO TO 5
      IF(R(I).EQ.0.AND.R(J).EQ.0.AND.R(K).EQ.1.AND.PS(I).EQ.2) GO TO 6
      IF(R(I).EQ.0.AND.R(J).EQ.0.AND.R(K).EQ.1.AND.PS(I).EQ.3) GO TO 5
      IF(R(I).EQ.0.AND.R(J).EQ.0.AND.R(K).EQ.1.AND.PS(I).EQ.7) GO TO 5
      IF(R(I).EQ.0.AND.R(J).EQ.1.AND.PS(I).EQ.4) GO TO 7
      IF(R(I).EQ.0.AND.R(J).EQ.1.AND.PS(I).EQ.6) GO TO 7
      IF(R(I).EQ.0.AND.R(J).EQ.1.AND.PS(I).EQ.7) GO TO 7
      IF(R(I).EQ.1.AND.R(J).EQ.0.AND.PS(I).EQ.5) GO TO 8
      IF(R(I).EQ.1.AND.R(J).EQ.0.AND.PS(I).EQ.9) GO TO 8
      IF(R(I).EQ.1.AND.R(J).EQ.0.AND.PS(I).EQ.8) GO TO 9
      IF(R(I).EQ.1.AND.R(J).EQ.1.AND.PS(I).EQ.5) GO TO 10
      IF(R(I).EQ.1.AND.R(J).EQ.1.AND.PS(I).EQ.9) GO TO 10
      IF(R(I).EQ.1.AND.R(J).EQ.1.AND.PS(I).EQ.8) GO TO 11
    2 A1(I)=0; A2(I)=0; PS(J)=1
      GO TO 1
    3 A1(I)=1; A2(I)=0; PS(J)=2
      GO TO 1
    4 A1(I)=0; A2(I)=1; PS(J)=3
      GO TO 1
    5 A1(I)=0; A2(I)=0; PS(J)=4
      GO TO 1
    6 A1(I)=0; A2(I)=1; PS(J)=4
      GO TO 1
```

```
    7   A1(I)=0; A2(I)=0; PS(J)=5
        GØ TØ 1
    8   A1(I)=0; A2(I)=1; PS(J)=6
        GØ TØ 1
    9   A1(I)=0; A2(I)=0; PS(J)=7
        GØ TØ 1
   10   A1(I)=1; A2(I)=0; PS(J)=8
        GØ TØ 1
   11   A1(I)=0; A2(I)=0; PS(J)=9
    1   CØNTINUE
        WRITE(6,172)
 172     FØRMAT(1H1)
        DØ 80  I=1,NBITS/100+1
        I1=(I-1)*100+1
        I2=I1+99
        I2=MIN(I1+99,NBITS)
        WRITE(6,85)  ( R(J),J=I1,I2)
        WRITE(6,85)  (A1(J),J=I1,I2)
        WRITE(6,85)  (A2(J),J=I1,I2)
 85     FØRMAT(' ',100I1)
   80   CØNTINUE
   12   CØNTINUE
        RETURN
        END
```

LEVEL I DECODE

```
      SUBRØUTINE DECØDE(NBITS,R   )
      IMPLICIT INTEGER (A-Z)
      CØMMØN  A1(640),A2(640),PS(640),DR(640),RØ(640)
      DIMENSIØN BM(9,400),HD(9,400),          RD(400),DBM(9,400)
      DIMENSIØN B1(400),B2(400)
      DIMENSIØN R(1)
      N=16
      EDIT=0
      DIMENSIØN P(200),PP(200),PM(9,200)
      P(1)=P(2)=0
  200 FØRMAT(I3)
      HD(1,1)=HD(2,1)=HD(3,1)=HD(4,1)=HD(5,1)=HD(6,1)=HD(7,1)=HD(8,1)=HD
     1(9,1)=0
      HD(1,2)=HD(2,2)=HD(3,2)=HD(4,2)=HD(5,2)=HD(6,2)=HD(7,2)=HD(8,2)=HD
     1(9,2)=0
      HD(2,3)=HD(3,3)=HD(7,3)=HD(9,3)=50
      SUM1=A1(1)+1
      IF(SUM1.EQ.2) SUM1=0
      SUM2=A2(1)+1
      IF(SUM2.EQ.2) SUM2=0
      SUM3=A1(2)+1
      IF(SUM3.EQ.2) SUM3=0
      SUM4=A2(2)+1
      IF(SUM4.EQ.2) SUM4=0
      SUM5=A1(3)+1
      IF(SUM5.EQ.2) SUM5=0
      SUM6=A2(3)+1
      IF(SUM6.EQ.2) SUM6=0
      HD(1,3)=SUM1  +A2(1)+A1(2)+SUM4  +A1(3)+A2(3)
      HD(1,3)=SUM1  +A2(1)+A1(2)+SUM4  +A1(3)+A2(3)
      HD(4,3)=SUM1  +A2(1)+A1(2)+SUM4  +A1(3)+A2(3)
      HD(5,3)=SUM1  +A2(1)+A1(2)+SUM4  +A1(3)+A2(3)
      HD(6,3)=A1(1)+A2(1)+A1(2)+A2(2)+A1(3)+SUM6
      HD(8,3)=A1(1)+A2(1)+A1(2)+A2(2)+SUM5  +A2(3)
      P(3)=MIN0(HD(1,3),HD(4,3),HD(5,3),HD(6,3),HD(8,3))
      BM(1,1)=BM(1,2)=BM(1,3)=0
      BM(4,1)=BM(4,2)=BM(4,3)=0
      BM(5,1)=BM(5,2)=0; BM(5,3)=1
      BM(6,1)=BM(6,2)=0; BM(6,3)=1
      BM(8,1)=0; BM(8,2)=BM(8,3)=1
      BM(2,1)=BM(2,2)=BM(2,3)=0
      BM(3,1)=BM(3,2)=BM(3,3)=0
      BM(7,1)=BM(7,2)=BM(7,3)=0
      BM(9,1)=BM(9,2)=BM(9,3)=0
```

```
      DØ 14 I=4,NBITS
      L=I-1
      SUM1=A1(I)+1
      IF(SUM1.EQ.2) SUM1=0
      SUM2=A2(I)+1
      IF(SUM2.EQ.2) SUM2=0
      HD1 = HD(6,L)+A1(I)+A2(I)
      HD2 = HD(3,L)+A1(I)+A2(I)
      HD3 = HD(1,L)+SUM1 +A2(I)
      HD4 = HD(7,L)+SUM1 +A2(I)
      HD5 = HD(2,L)+A1(I)+SUM2
      HD6 = HD(6,L)+A1(I)+A2(I)
      HD7 = HD(1,L)+A1(I)+A2(I)
      HD8 = HD(2,L)+A1(I)+SUM2
      HD9 = HD(3,L)+A1(I)+A2(I)
      HD10= HD(7,L)+A1(I)+A2(I)
      HD11= HD(4,L)+A1(I)+A2(I)
      HD12= HD(6,L)+A1(I)+A2(I)
      HD13= HD(7,L)+A1(I)+A2(I)
      HD14= HD(5,L)+A1(I)+SUM2
      HD15= HD(9,L)+A1(I)+SUM2
      HD16= HD(8,L)+A1(I)+A2(I)
      HD17= HD(5,L)+SUM1 +A2(I)
      HD18= HD(9,L)+SUM1 +A2(I)
      HD19= HD(8,L)+A1(I)+A2(I)
      IF (EDIT.EQ.1)
     .PRINT 15,HD1,HD2,HD3,HD4,HD5,HD6,HD7,HD8,HD9,HD10,HD11,HD12,HD13,H
     1D14,HD15,HD16,HD17,HD18,HD19
   15 FØRMAT(20I4)
      Q=MIN0(HD1,HD2)
      M1=1
      M2=L
      IF(I.GT.N) M1=I-N+1
      DØ 16 M=M1,M2
      DBM(1,M)=BM(1,M)
      DBM(2,M)=BM(2,M)
      DBM(3,M)=BM(3,M)
      DBM(4,M)=BM(4,M)
      DBM(5,M)=BM(5,M)
      DBM(6,M)=BM(6,M)
      DBM(7,M)=BM(7,M)
      DBM(8,M)=BM(8,M)
        DBM(9,M)=BM(9,M)
   16 CØNTINUE
      IF (EDIT .NE. 1)  GØ TØ 171
      DØ 17 NDX1=1,9
      PRINT 18,I,(BM(NDX1,NDX2), NDX2=M1,M2)
   18 FØRMAT(I40,10X,19I3)
   17 CØNTINUE
  171 CØNTINUE
      M2=I-1
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      L=I
      IF(Q.EQ.HD1)  GØ TØ 19
      IF(Q.EQ.HD2)  GØ TØ 20
19    HD(1,L)=HD1; BM(1,L)=0
      DØ 38 V=M1,M2
      BM(1,V)=DBM(6,V)
38    CØNTINUE
      GØ TØ 57
20    HD(1,L)=HD2; BM(1,L)=0
      DØ 39 V=M1,M2
      BM(1,V)=DBM(3,V)
39    CØNTINUE
57    Q=MIN0(HD3,HD4)
      IF(Q.EQ.HD3)  GØ TØ 21
      IF(Q.EQ.HD4)  GØ TØ 22
21    HD(2,L)=HD3; BM(2,L)=0
      DØ 40 V=M1,M2
      BM(2,V)=DBM(1,V)
40    CØNTINUE
      GØ TØ 58
22    HD(2,L)=HD4; BM(2,L)=0
      DØ 41 V=M1,M2
      BM(2,V)=DBM(7,V)
41    CØNTINUE
58    Q=MIN0(HD6,HD7,HD8,HD9,HD10)
      HD(3,L)=HD5; BM(3,L)=0
      DØ 42 V=M1,M2
      BM(3,V)=DBM(2,V)
42    CØNTINUE
      IF(Q.EQ.HD6)  GØ TØ 24
      IF(Q.EQ.HD7)  GØ TØ 25
      IF(Q.EQ.HD8)  GØ TØ 26
      IF(Q.EQ.HD9)  GØ TØ 27
      IF(Q.EQ.HD10) GØ TØ 28
24    HD(4,L)=HD6; BM(4,L)=0
      DØ 43 V=M1,M2
      BM(4,V)=DBM(6,V)
43    CØNTINUE
      GØ TØ 59
25    HD(4,L)=HD7; BM(4,L)=0
      DØ 44 V=M1,M2
      BM(4,V)=DBM(1,V)
44    CØNTINUE
      GØ TØ 59
26    HD(4,L)=HD8; BM(4,L)=0
      DØ 45 V=M1,M2
      BM(4,V)=DBM(2,V)
45    CØNTINUE
      GØ TØ 59
27    HD(4,L)=HD9; BM(4,L)=0
      DØ 46 V=M1,M2
```

```
      BM(4,V)=DBM(3,V)
   46 CØNTINUE
      GØ TØ 59
   28 HD(4,L)=HD10; BM(4,L)=0
      DØ 47 V=M1,M2
      BM(4,V)=DBM(7,V)
   47 CØNTINUE
   59 Q=MIN0(HD11,HD12,HD13)
      IF(Q.EQ.HD11) GØ TØ 29
      IF(Q.EQ.HD12) GØ TØ 30
      IF(Q.EQ.HD13) GØ TØ 31
   29 HD(5,L)=HD11; BM(5,L)=1
      DØ 48 V=M1,M2
      BM(5,V)=DBM(4,V)
   48 CØNTINUE
      GØ TØ 60
   30 HD(5,L)=HD12; BM(5,L)=1
      DØ 49 V=M1,M2
      BM(5,V)=DBM(6,V)
   49 CØNTINUE
      GØ TØ 60
   31 HD(5,L)=HD13; BM(5,L)=1
      DØ 56 V=M1,M2
      BM(5,V)=DBM(7,V)
   56 CØNTINUE
   60 Q=MIN0(HD14,HD15)
      IF(Q.EQ.HD14) GØ TØ 32
      IF(Q.EQ.HD15) GØ TØ 33
   32 HD(6,L)=HD14; BM(6,L)=0
      DØ 50 V=M1,M2
      BM(6,V)=DBM(5,V)
   50 CØNTINUE
      GØ TØ 61
   33 HD(6,L)=HD15; BM(6,L)=0
      DØ 51 V=M1,M2
      BM(6,V)=DBM(9,V)
   51 CØNTINUE
   61 Q=MIN0(HD17,HD18)
      HD(7,L)=HD16; BM(7,L)=0
      DØ 52 V=M1,M2
      BM(7,V)=DBM(8,V)
   52 CØNTINUE
      IF(Q.EQ.HD17) GØ TØ 35
      IF(Q.EQ.HD18) GØ TØ 36
   35 HD(8,L)=HD17; BM(8,L)=1
      DØ 53 V=M1,M2
      BM(8,V)=DBM(5,V)
   53 CØNTINUE
      GØ TØ 37
   36 HD(8,L)=HD18; BM(8,L)=1
      DØ 54 V=M1,M2
```

```
          BM(8,V)=DBM(9,V)
   54     CØNTINUE
   37     HD(9,L)=HD19; BM(9,L)=1
          DØ 55 V=M1,M2
          BM(9,V)=DBM(8,V)
   55     CØNTINUE
          IF(I.LT.N) GØ TØ 14
          J=I
          K=I-N+1
          Q=MINO(HD(1,J),HD(2,J),HD(3,J),HD(4,J),HD(5,J),HD(6,J),HD(7,J),HD(
         18,J),HD(9,J))
          IF(Q.EQ.HD(1,J)) GØ TØ 62
          IF(Q.EQ.HD(2,J)) GØ TØ 63
          IF(Q.EQ.HD(3,J)) GØ TØ 64
          IF(Q.EQ.HD(4,J)) GØ TØ 65
          IF(Q.EQ.HD(5,J)) GØ TØ 66
          IF(Q.EQ.HD(6,J)) GØ TØ 67
          IF(Q.EQ.HD(7,J)) GØ TØ 68
          IF(Q.EQ.HD(8,J)) GØ TØ 69
          IF(Q.EQ.HD(9,J)) GØ TØ 70
   62     RØ(K)=BM(1,K)
          GØ TØ 71
   63     RØ(K)=BM(2,K)
          GØ TØ 71
   64     RØ(K)=BM(3,K)
          GØ TØ 71
   65     RØ(K)=BM(4,K)
          GØ TØ 71
   66     RØ(K)=BM(5,K)
          GØ TØ 71
   67     RØ(K)=BM(6,K)
          GØ TØ 71
   68     RØ(K)=BM(7,K)
          GØ TØ 71
   69     RØ(K)=BM(8,K)
          GØ TØ 71
   70     RØ(K)=BM(9,K)
   71     CØNTINUE
   14     CØNTINUE
          IF (EDIT .NE. 1)  GØ TØ 721
          DØ 72 I=1,NBITS
          PRINT 73,I,BM(1,I),BM(2,I),BM(3,I),BM(4,I),BM(5,I),BM(6,I),BM(7,I)
         1,BM(8,I),BM(9,I),HD(1,I),HD(2,I),HD(3,I),HD(4,I),HD(5,I),HD(6,I),H
         2D(7,I),HD(8,I),HD(9,I)
   73     FØRMAT(19I5)
   72     CØNTINUE
  721     CØNTINUE
          WRITE(6,172)
  172     FØRMAT(1H1)
          DØ 76 I=1,NBITS
          R(I)=R(I+3)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
          RØ(I)=RØ(I+2)
   76     CØNTINUE
          DØ 74 I=1,NBITS
          RD(I)=RØ(I)-R(I)
   74     CØNTINUE
          DØ 741 I=1,NBITS/100+1
          I1=(I-1)*100+1
          I2=I1+99
          I2=MIN(I1+99,NBITS)
          WRITE(6,85)  (RØ(J),J=I1,I2)
          WRITE(6,85)  (R (J),J=I1,I2)
          WRITE(6,85)  (RD(J),J=I1,I2)
 85       FØRMAT(' ',100I1)
 741      CØNTINUE
          RETURN
          END
```