

AD-786 492

MACHINE RECOGNITION OF HAND-SENT MORSE
CODE USING THE PDP-12 COMPUTER

Joel Arthur Guenther

Air Force Institute of Technology
Wright-Patterson Air Force Base, Ohio

December 1973

DISTRIBUTED BY:

NTIS

National Technical Information Service
U. S. DEPARTMENT OF COMMERCE
5285 Port Royal Road, Springfield Va. 22151

MACHINE RECOGNITION OF
HAND-SENT MORSE CODE
USING THE PDP-12 COMPUTER

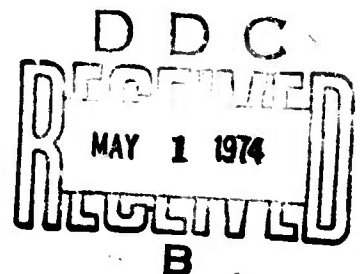
THESIS

GE/EE/73A-9

Joel A. Guenther
Captain USAF

This document has been approved for public release and sale;
its distribution is unlimited.

ib



Unclassified

Security Classification

AD786492

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Air Force Institute of Technology (AFIT-EN) ✓ Wright-Patterson AFB, Ohio 45433		2a. REPORT SECURITY CLASSIFICATION Unclassified	
		2b. GROUP	
3. REPORT TITLE Machine Recognition of Hand-Sent Morse Code using the PDP-12 Computer ✓			
4. DESCRIPTIVE NOTES (Type of report and inclusive dates) AFIT Thesis			
5. AUTHOR(S) (First name, middle initial, last name) Joel A. Guenther Captain USAF			
6. REPORT DATE December 1973		7a. TOTAL NO. OF PAGES 153	7b. NO. OF REFS 8
8a. CONTRACT OR GRANT NO.		9a. ORIGINATOR'S REPORT NUMBER(S) GE/EE/73A-9 ✓	
b. PROJECT NO.		9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	
c.			
d.			
10. DISTRIBUTION STATEMENT Approved for public release; distribution unlimited			
11. SUPPLEMENTARY NOTES Approved for public release; IAW AFR 190-17. JERRY C. HIX, Captain, USAF Director of Information		12. SPONSORING MILITARY ACTIVITY 6570th AMRL Wright-Patterson AFB, OH	
13. ABSTRACT The purpose of this investigation is to determine an optimum decision algorithm for use in machine recognition of hand-sent Morse code. An extensive analysis of hand-sent Morse code data is presented together with a discussion on the relative merits of several recognition algorithms. A recognition program is developed for use on the PDP-12 digital computer to test these algorithms. Test results are presented for a time duration averaging algorithm which achieves less than a one per cent recognition error rate for noise-free Morse code signals.			

Reproduced for
NATIONAL TECHNICAL
INFORMATION SERVICE
U S Department of Commerce
Springfield VA 22151

166

DD FORM 1 NOV 65 1473

Unclassified

Security Classification

Unclassified
Security Classification

[illegible]

Unclassified
Security Classification

MACHINE RECOGNITION OF HAND-SENT
MORSE CODE USING THE PDP-12 COMPUTER

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
in Partial Fulfillment of the
Requirements for the Degree of
Master of Science

by

Joel A. Guenther, BSEE

Captain USAF

Graduate Electrical Engineering

December 1973

This document has been approved for public release and sale;
its distribution is unlimited.

ic

Contents

	Page
Preface.....	iv
List of Figures.....	v
List of Tables.....	viii
Abstract.....	ix
I. Introduction.....	1
II. Properties of Hand-Sent Morse Code.....	4
International Morse Code.....	4
Code Sending Instruments.....	5
Characteristics of Hand-Sent Morse Code.....	6
III. Data Analysis Procedure.....	8
Data Gathering.....	8
Data Categorizing.....	9
Analog-to-Digital Conversion.....	9
Manual Identification.....	10
Distribution Plots.....	12
Data Analysis.....	12
Pulse Algorithms.....	14
Space Algorithms.....	16
IV. The Computer Recognition Program.....	23
Program Constraints.....	23
PDP-12 Description.....	23
Operational Goals.....	26
Program Description.....	27
Signal Processing Section.....	31
Signal Sampling Routine.....	33
Change Detection Routine.....	33
Code Translation Section.....	38
Initialization Routine.....	40
Data Recognition Routine.....	44
Internal Code Generation Routine.....	48
Character Identification Routine.....	50
Word Space Correction Routine.....	54
Error Correction Routine.....	56
V. Operational Procedure.....	60
System Initiation.....	60
Real Time Adjustments.....	62
Restart Procedure.....	63

Contents

	Page
VI. Results.....	65
Man vs. Machine Comparison.....	65
Testing Procedure.....	66
Classification of Errors.....	67
Analysis of Performance.....	71
Error Rates.....	72
Limitations.....	72
VII. Conclusions and Recommendations.....	75
Conclusions.....	75
Recommendations.....	76
Recognition Program Improvements.....	76
Hardware Realization.....	78
Bibliography.....	80
Appendix A: Recognition Program Listing.....	81
Appendix B: Hand-Sent Morse Code Data Plots.....	113
Appendix C: Code and Character Listing.....	129
Appendix D: Computer Test Messages.....	133
Vita.....	153

Preface

This thesis is the result of an effort to provide real-time machine recognition of hand-sent Morse code through the use of a minicomputer. While the capability to recognize hand-sent Morse code messages by machine has been demonstrated before on large scale special purpose computers, the main contribution of this study was to do it with a relatively inexpensive general purpose minicomputer.

I wish to express my gratitude to my advisor, Lt. Col. Tom Purnhagen, for his assistance and guidance throughout the development of this study. I wish also to express my thanks to Captain Joseph Carl, of the Aerospace Medical Research Laboratory, for his appreciated support of this project as my laboratory sponsor, and to the Dayton Amateur Radio Association for their assistance in obtaining Morse code transmissions for use in this project. Special thanks are reserved for my wife and family for their sacrifices and patience while I spent my evenings conversing with the computer.

Joel A. Guenther

List of Figures

Figure		Page
3-1	CRT Morse Code Display.....	11
3-2	Linear Decision Boundaries for Recording Session 1...	18
3-3	Linear Decision Boundaries for Recording Session 2...	19
3-4	Linear Decision Boundaries for Recording Session 3...	20
4-1	Signal Processing Section Organization.....	28
4-2	Code Translation Section Organization.....	28
4-3	Service Program Flowchart.....	30
4-4	Signal Sampling Routine Flowchart.....	34
4-5	Change Detection Routine Flowchart.....	35
4-6	Initialization Routine Flowchart.....	41
4-7	Data Recognition Routine Flowchart.....	45
4-8	Internal Code Generation Routine Flowchart.....	49
4-9	Internal Code Word Generation.....	50
4-10	Character Identification Routine Flowchart.....	51
4-11	Word Space Correction Routine Flowchart.....	55
4-12	Error Correction Routine Flowchart.....	57
5-1	Recognition Program Operational Configuration.....	61
B-1	Morse Code Data Distribution Plot, DOT (Only) Time Duration vs. Time Duration of Following Space (Recording Session 1).....	114
B-2	Morse Code Data Distribution Plot, DOT (First) Time Duration vs. Time Duration of Following Space (Recording Session 1).....	115
B-3	Morse Code Data Distribution Plot, DOT (Intermediate) Time Duration vs. Time Duration of Following Space (Recording Session 1).....	116
B-4	Morse Code Data Distribution Plot, DOT (Last Character) Time Duration vs. Time Duration of Following Space (Recording Session 1).....	117

List of Figures

Figure		Page
B-5	Morse Code Data Distribution Plot, DOT (Last Word) Time Duration vs. Time Duration of Following Space (Recording Session 1).....	118
B-6	Morse Code Data Distribution Plot, DASH (Only) Time Duration vs. Time Duration of Following Space (Recording Session 1).....	119
B-7	Morse Code Data Distribution Plot, DASH (First) Time Duration vs. Time Duration of Following Space (Recording Session 1).....	120
B-8	Morse Code Data Distribution Plot, DASH (Intermediate) Time Duration vs. Time Duration of Following Space (Recording Session 1).....	121
B-9	Morse Code Data Distribution Plot, DASH (Last Character) Time Duration vs. Time Duration of Following Space (Recording Session 1).....	122
B-10	Morse Code Data Distribution Plot, DASH (Last Word) Time Duration vs. Time Duration of Following Space (Recording Session 1).....	123
B-11	Morse Code Data Distribution Plot, Pulse Time Duration vs. Time Duration of Following Space (Recording Session 1).....	124
B-12	Morse Code Data Distribution Plot, Pulse Time Duration vs. Time Duration of Following Space (Recording Session 2).....	125
B-13	Morse Code Data Distribution Plot, Pulse Time Duration vs. Time Duration of Following Space (Recording Session 3).....	126
B-14	DASH (Last Character) and Following Space Time Duration Fluctuations (Recording Session 1).....	127
B-15	DASH (Last Character) and Following Space Time Duration Fluctuations (Recording Session 2).....	128
D-1	Prepared Text for Recording Sessions 1 through 4.....	134
D-2	Recognition Program Output for Recording Session 1.....	136
D-3	Recognition Program Output for Recording Session 2.....	139
D-4	Recognition Program Output for Recording Session 3.....	142

List of Figures

Figure		Page
D-5	Recognition Program Output for Recording Session 4.....	145
D-6	Recognition Program Output for Recording Session 5.....	148
D-7	Recording Session 6 Text (WLAW Bulletin).....	150
D-8	Recognition Program Output for Recording Session 6.....	151
D-9	Recognition Program Output for Recording Session 7.....	152

List of Tables

Table		Page
I	Pulse and Space Categories.....	10
II	Cluster Identification for Figures 3-2, 3-3, and 3-4.....	17
III	Recognition Program Real-Time Adjustments.....	63
IV	Recording Session Variables and Error Rates.....	68

Abstract

The purpose of this investigation is to determine an optimum decision algorithm for use in machine recognition of hand-sent Morse code. An extensive analysis of hand-sent Morse code data is presented together with a discussion on the relative merits of several recognition algorithms. A recognition program is developed for use on the PDP-12 digital computer to test these algorithms. Test results are presented for a time duration averaging algorithm which achieves less than a one per cent recognition error rate for noise-free Morse code signals.

MACHINE RECOGNITION OF
HAND-SENT MORSE CODE
USING THE PDP-12 COMPUTER

I. Introduction

The idea of machine recognition of Morse code is not a new one. Much work has been done to develop equipment for the automatic reception of Morse code signals in the last two decades. The first notable effort was undertaken by Lincoln Laboratories at the Massachusetts Institute of Technology in the late 1950s (Ref 4). The machine which resulted from this project was actually a special purpose digital computer, called MAUDE (Morse AUtomatic DEcoder). The recognition algorithm used in MAUDE was based on its "knowledge" of Morse code linguistic properties and of the relative time durations of marks (referred to as pulses throughout this report) and spaces. MAUDE demonstrated a 90 to 95 per cent correct decoding rate, although this rate was later improved by the addition of an output error detection and correction scheme. The main disadvantage of MAUDE, however, was its physical size and complexity.

The development of small, low-cost integrated circuits led to the design and fabrication of several Morse-to-teletype converters. Generally speaking, much simpler code recognition algorithms, as compared to those used in MAUDE, were used in these converters to identify pulses and spaces. One such converter, designed at the Naval Postgraduate School in 1968 (Ref 6), uses the time duration of the most recently received short pulse (DOT) as a reference unit for pulse and space identification. Multiples of this reference unit are used as decision thresholds to identify succeeding pulses and spaces, and to eliminate signal noise. Another

example of this type of converter, called the Morse-A-Verter, uses a variation of the previous recognition algorithm to make pulse-space decisions (Ref 5). The time duration of the most recently processed pulse, short or long, is used to determine the classification of the succeeding pulse. Spaces are identified by comparison with the time duration of the most recent long pulse (DASH).

Morse code recognition machines perform five basic functions. First, the pulse-modulated audio Morse signal is converted into a form usable by the machine, usually dc pulses, while discriminating against noise. Second, the time duration and identification of each pulse and space is determined. Third, pulses and spaces are classified into one of two pulse or three space categories according to their relative time duration. Fourth, the categorized pulses and spaces are combined to form Morse code characters. Finally, a signal representing the identified Morse code character is transmitted to an output device.

The third function, that of categorizing pulses and spaces, is the most difficult to perform. This difficulty is mainly due to the inherently non-uniform pulse and space time durations of hand-sent code. Since these time durations form the basis for the recognition process, a rigid set of decision algorithms, such as those used in commercial Morse telegraphy equipment, cannot be used. Instead, algorithms based on traits common to all variations of hand-sent Morse code must be used to achieve the highest possible degree of machine recognition accuracy.

The objective of this study is two fold: first, to conduct a thorough examination of hand-sent Morse code data and identify common traits which may be used in a machine recognition process, and second, to develop a Morse code recognition program, for use on the PDP-12 digital

computer, to test and refine decision algorithms based on these common traits. Thus, the main concern here is optimization of the third function performed by Morse code recognition machines, as previously defined. Of course, all five functions must be considered in the development of the PDP-12 computer program. The remaining four functions, however, are of secondary concern to this project.

A discussion on the properties of hand-sent Morse code, and on the primary factors responsible for the variations present in hand-sent Morse code, is presented in Chapter II. Chapter III describes the procedure used to obtain and analyze Morse code data, and the common traits discovered during and the decision algorithms derived from this analysis procedure. A complete description of the resulting computer recognition program is presented in Chapter IV, as well as a brief description of the PDP-12 computer and peripheral devices used in this project. The operational procedure used with the recognition program is described in Chapter V. Chapter VI presents an analysis of the results obtained during the testing procedure. Conclusions and recommendations are contained in Chapter VII.

II. Properties of Hand-Sent Morse Code

This chapter presents a discussion on the characteristic properties of hand-sent Morse code. This discussion includes a definition of international Morse code, a description of common mechanical devices used to transmit Morse code, and the problems associated with machine recognition of hand-sent Morse code.

International Morse Code

Morse code is a rudimentary one-dimensional binary encoding scheme for language in which each character is represented by a unique sequence of pulses and spaces. These characters represent letters, numbers, punctuation signs, and special symbols. The international Morse code alphabet is given in Appendix C.

Two types of pulses and three types of spaces, distinguished by their relative time durations, are used to define Morse code characters. In terms of time units, the accepted standard definitions for these pulses and spaces are: pulses- DOT = 1 and DASH = 3; spaces- SYMBOL = 1, CHARACTER = 3, and WORD = 5 to 7. Characters are defined by a sequence of pulses separated by spaces. SYMBOL spaces separate pulses within a character, CHARACTER spaces separate characters within words, and WORD spaces separate words.

Morse code transmission speed is measured in terms of words per minute (wpm), with an average of 5 characters per word assumed as standard. Transmission rates for hand-sent Morse code are normally in the 10 to 40 wpm range, although rates as high as 60 wpm are not uncommon. Morse code machines are capable of operating at much faster rates, but generally do not unless another machine is used to receive the transmitted message.

Code Sending Instruments

Morse code is transmitted by keying an oscillator tuned to the desired transmission frequency or subharmonic of this frequency. The four standard instruments (keys) used to generate Morse code are, in ascending order of sophistication: 1) the simple hand key, 2) the semi-automatic key, or "bug", 3) the electronic keyer, and 4) the fully automatic Morse machine. A brief description of each instrument is presented in the following paragraphs.

The hand key, because of its simplicity and low cost, is most often used. Pulses are transmitted by depressing a paddle key; spaces are produced by lifting the key. The relative time duration in either position determines the type of pulse or space transmitted. The durations of all pulses and spaces are controlled directly by the sender.

The semiautomatic key, or "bug", is more difficult to operate and is generally used by the more experienced operator. Two paddle keys are used, one for DOTs and one for DASHes. The DOT key is used to produce a machinelike sequence of alternating DOTs and SYMBOL spaces for as long as the key is depressed. The DASH key produces DASHes in the same manual fashion as is done on the hand key.

The electronic keyer produces regulated DOTs, DASHes, and SYMBOL spaces. Two paddle keys are again used, one for DOTs and one for DASHes. Either key generates a sequence of pulses and SYMBOL spaces for as long as it is depressed. CHARACTER and WORD spaces, however, are still controlled manually. The time durations of automatically generated pulses and spaces can be adjusted to match the sending rate of the particular transmission.

Fully automatic Morse machines regulate the time durations of all

pulses and spaces. Messages are prepared ahead of time on paper tape or stored in a memory device for transmission by the machine. The transmission and reception of completely automatic Morse code is not of concern in this project.

Characteristics of Hand-Sent Morse Code

The time duration standards for Morse code pulse and space relationships presented earlier are not always realized in hand-sent transmissions. Commonly encountered distortions include fluctuating pulse and space time duration ratios and variations in sending speed. In hand-sent Morse code, the time durations of pulse and space elements vary substantially from their prescribed values, and recognition must be based on the proportions of the time durations of these elements. These proportions generally vary non-trivially.

In a 1968 report on Morse code teaching methods (Ref 8), it was noted that the chief type of code reception error arises from the tendency to hear code signals shorter than they really are. For example, five DOTs are heard as four, four DOTs as three, etc. The tendency toward "signal shrinkage" mainly involves the last element of a code character. The report further states that this tendency to hear signals shorter than they are may be responsible for the general tendency for all operators to lengthen terminal DASHes. Also noted was the tendency of operators at all levels of skill to make pulse and space time duration ratios larger than the theoretical 1:3:7 ratios. However, a better set of ratios could not be recommended since actual ratios varied widely between individual operators.

Transmission rates generally tend to decrease over an extended period of time. This is mainly due to physical as well as mental fatigue.

Pulse and space ratios also tend to change as the transmission rate decreases. The amount of ratio change varies from sender to sender, and is partly a function of the type of sending unit being used. When hand keys are used, slower transmission rates generally result in longer pulses and spaces, with spaces tending to lengthen proportionately more than pulses. The use of semiautomatic keys prevents the automatically generated pulses and spaces from being affected by speed variations. Only those output functions controlled by the operator are subject to change. Thus, the degree of ratio change due to speed change may be significantly different when semiautomatic keys are used than when hand keys are used.

The recognition algorithms used in the machines discussed earlier may not be optimal. The prospects for the useful application of linguistic techniques directly to binarised hand-sent Morse code do not appear to be good because of the non-trivial variation of element proportions (Ref 7:254). Decisions made strictly on an element-to-element basis may not be flexible enough to cope with the widely varying proportions found in hand-sent transmissions.

These variations in hand-sent Morse code transmissions are the crux of the machine recognition problem. Code recognition machine performance ultimately depends on the algorithms used to identify individual pulse and space characters. Many algorithms that work well with certain Morse signals perform miserably with others. An algorithm is needed that can correctly identify pulse and space characters for all possible hand-sent Morse code variations.

III. Data Analysis Procedure

Many possible decision algorithms may be used to recognize hand-sent Morse code by machine. Indeed, each of the three recognition machines discussed in Chapter I employs a different method to perform the recognition process. Since there was no intuitively "best" method to use in performing this process, a thorough data analysis procedure was undertaken in search for an "optimal" decision algorithm, i.e., one that would yield the smallest percentage of recognition errors for all types of Morse code transmissions. The procedures used to obtain and analyze hand-sent Morse code data and the decision algorithms derived from this analysis are presented in the following paragraphs.

Data Gathering

Three samples of Morse code transmissions were recorded for analysis. These samples differ from each other in two ways: 1) the type of sending unit used, and 2) the degree of operator proficiency. The first sample, Recording Session 1, was transmitted with a "bug" at a rate of approximately 15 words per minute. Recording Session 2 was transmitted with a hand key at a rate of approximately 10-12 words per minute. Recording Session 3 was transmitted with a "bug" at a rate of approximately 18-20 words per minute. The three recording sessions were transmitted by separate individuals at their normal speed. None of these individuals were, at the time of the recordings, actively involved in cw (continuous wave) transmissions as a hobby, although they were at some time in their past. Thus, the recordings, are biased towards a low-proficiency level. As pointed out in the next section, this resulted in a slightly erratic sending rate and a wide spread of pulse and space time durations over a given time period.

The recording sessions were conducted in the following manner. A 500-word text, taken from The Radio Amateur's Handbook (Ref 1:7), was prepared for use as the message to be transmitted. A copy of this message, listed in Appendix D, was given to each of three individuals. While one individual transmitted the message, the other two annotated observed sending errors on their copy of the text. The Morse code transmission was recorded for use in the data analysis process. Only one recording session was held at a time to prevent mental fatigue from adversely affecting both the sender and the receivers. Recording sessions were held every other day until complete. The annotated copies of the transmitted Morse code messages were saved for use in evaluating recognition program performance, as explained in Chapter VI.

Data Categorizing

The recorded hand-sent Morse code transmissions were examined in a three step process. First, the time durations of pulses and spaces were obtained and stored on magnetic computer tape. Second, each pulse and space was identified as belonging to one of twenty different categories and again stored on magnetic tape. Finally, the categorized pulses and spaces were plotted for visual examination. Each of these three steps will now be discussed.

Analog-to-Digital Conversion. The time duration for each transmitted pulse and space was obtained and stored through the use of the PDP-12 computer in the following manner. The recorded Morse code transmission was connected to an A-D Converter external input channel and sampled periodically. When a change was detected, i.e., pulse-to-space or space-to-pulse, the time duration indicated on a real-time clock was recorded

and stored. The clock was then reset and the process repeated.

Approximately the first 200 words of each 500-word Morse code transmission were processed in this manner. Upon completion of this process, the stored time durations were visually examined and categorized, as explained next.

Manual Identification. Pulses and spaces were classified into 10 separate categories each. Space categories were chosen to permit investigation of the pulse vs. following space interrelations known to exist in hand-sent Morse code. For example, the time duration of a space, when preceded by a DASH, is generally less than it is when preceded by a DOT. Thus, space categories correspond directly to the category of the preceding pulse. Pulses were separated into five DOT and five DASH categories according to their relative position within a Morse code character. In this way, time duration vs. position interrelationships, if any, would be disclosed. The 20 pulse and space categories are listed in Table I.

TABLE I

Pulse and Space Categories

1. DOT (Only)	11. DASH (Only)
2. Space following 1.	12. Space following 11.
3. DOT (First)	13. DASH (First)
4. Space following 3.	14. Space following 13.
5. DOT (Intermediate)	15. DASH (Intermediate)
6. Space following 5.	16. Space following 15.
7. DOT (Last Character)	17. DASH (Last Character)
8. Space following 7.	18. Space following 17.
9. DOT (Last Word)	19. DASH (Last Word)
10. Space following 9.	20. Space following 19.

The Only category identifies a DOT or DASH that by itself signifies a Morse code character (the characters E and T respectively). A pulse is First if it is the first pulse of at least two pulses comprising a character. Likewise, a pulse is Last if it appears as the last pulse in the character. The Last Character and Last Word categories are determined by the type of space following the pulse, i.e., CHARACTER space or WORD space. An Intermediate pulse is one which is neither first nor last in a multi-pulse string. Only category pulses which also appear as the last pulse in a word are categorized as Last Word.

The visual identification process was performed in the following manner. Stored pulse and space time durations were displayed on the PDP-12 CRT Display screen as a series of lines, proportionate in length to the time duration represented, as depicted in Fig. 3-1. A cursor was also displayed to indicate the particular pulse to be categorized. Note that it is only necessary to identify pulses, since spaces are categorized by the type of preceeding pulse. The pulse was then visually identified by noting its relative length and position with respect to surrounding pulses and spaces. The pulse and following space time durations were then stored according to their respective categories by depressing one

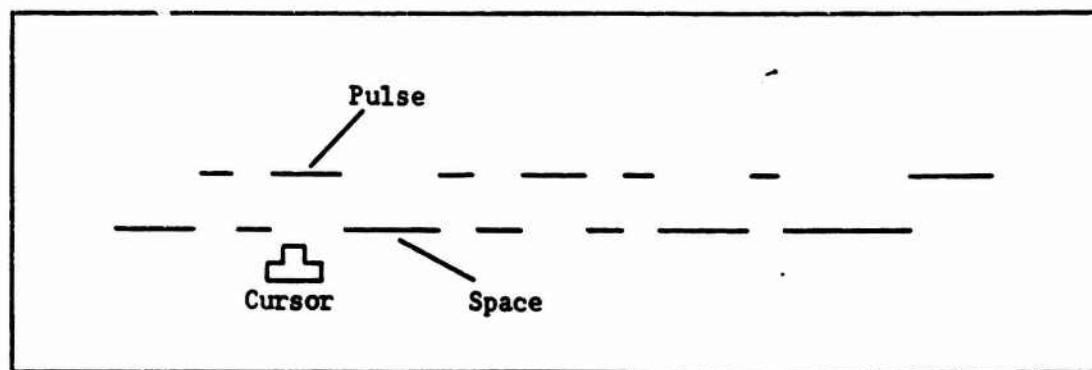


Fig. 3-1. CRT Morse Code Display

of 10 keys on a teletypewriter. Depression of the teletype key also advanced the display to the next pulse to be identified. The process was continued in this manner until all pulses and spaces were categorized and stored.

Distribution Plots. Cluster-type distribution plots of the categorized pulses and spaces were obtained by plotting pulse time duration versus following space time duration for the 10 categories. Plots were made of each individual category as well as all categories combined for the three recording sessions. The individual category plots for Recording Session 1 and combined plots for all recording sessions are contained in Appendix B, Figures B-1 through B-13.

Data Analysis

The individual and combined distribution cluster plots for the three recording sessions were examined to identify any possible relationships that might be used in the recognition program. Several such relationships were found.

It was immediately obvious that some of the pulse and following space categories are essentially identical and can be combined into one unique category. The First and Intermediate categories for both DOTs and DASHes are identical, as are the Only and Last Character categories. Thus, the 10 original categories can be reduced to 6 categories, 3 for each type of pulse. These three categories correspond to the type of following space, i.e., SYMBOL, CHARACTER, and WORD.

Another obvious trait disclosed by the data plots was the large variance of CHARACTER and WORD space time durations. Since the plots represent data obtained over an extended period of time (approximately

10-15 minutes) it was thought possible that the large variance was due to a gradual change in transmission speed during the time interval. To investigate this possibility, plots of pulse and space durations versus their sending sequence in time were made for the DASH Intermediate, Last Character, and Last Word categories. These plots, two of which are included in Appendix B (Figures B-14 and B-15), indicate that the large variance is not a function of a general speeding-up or slowing-down trend, but is, in fact, a characteristic of hand-sent Morse code.

Examination of the data plots for all categories combined indicates that a large overlap exists between CHARACTER space and WORD space time durations. The existence of this overlap prohibits correct identification of CHARACTER spaces versus WORD spaces on a time duration threshold basis. However, this distinction is not a critically important one, since both types of spaces signify the end of a Morse code character. The combined category data plots do, however, indicate a wide gap between SYMBOL space and non-SYMBOL space clusters. These gaps lend themselves to the formation of linear decision boundaries in two-dimensional pattern space quite easily.

Another obvious condition revealed by the combined data plots is the wide gap between DOT and DASH time durations. This again is conducive to a linear decision technique.

The combined data plots reveal two distinct correlations between DASH time durations and the type of following space. The first, and most obvious of the two is that DASH time durations are generally longer when followed by a CHARACTER or WORD space than they are when followed by a SYMBOL space. The second correlation is that SYMBOL space durations tend to decrease as the time duration of the preceding DASH increases,

and vice versa. Neither of these pulse-space correlations are evident in the DOT-space categories.

As discussed earlier in Chapters I and III, several different types of decision algorithms have been successfully used to recognize hand-sent Morse code. Most notable among these are the use of a unique set of linguistic rules and the comparison of the time duration of the previous pulse with that of the next space and pulse on a threshold decision basis. A different approach to the recognition problem was looked for in this project; one that might prove more successful than those methods previously tried.

Evaluation of all the observations made from the data distribution plots led to the derivation of pulse and space linear decision algorithms, based on time duration averages, for use in the recognition program. These algorithms are discussed in the following paragraphs.

Pulse Algorithms. Due to the wide separation between DOT and DASH time durations, and the small variance of these durations, a linear decision boundary can easily be established as a function of DOT and DASH averages. However, to allow for slowly changing transmission rates, and to suppress temporarily large excursions from the mean, individual DOT and DASH averages must be calculated on a floating basis. That is, the averages must be computed for the last N DOTs and DASHes received, rather than on all received since the start of Morse code processing. The optimum size of N is that which both suppresses large excursions and permits the average to follow slowly varying changes. A value of eight achieves these goals and, as discussed below, permits easy average computation on the PDP-12 computer.

The DOT average is computed by the following equation:

$$\text{DOT AVG.} = \text{DOT AVG.} + \frac{\text{NEW DOT}}{8} - \frac{\text{DOT AVG.}}{8} \quad (3.1)$$

When a newly received pulse is identified as a DOT, the DOT average is recomputed to include the new time duration information. The division process is performed by shifting the 12-bit register, containing the quantity to be divided, three places to the right. This is the equivalent of dividing the quantity by 2^3 or 8. The use of this process eliminates the time consuming task of adding 8 registers together and then dividing by 8. The DASH average is computed in the same manner as the DOT average.

The pulse average is computed after each recomputation of either the DOT average or the DASH average by the following equation:

$$\text{PULSE AVG.} = \frac{\text{DOT AVG.}}{4} + \frac{\text{DASH AVG.}}{2} \quad (3.2)$$

The pulse average is used as the pulse time duration linear decision boundary. Note that the pulse average is not the mean of the DOT and DASH averages, but is instead, slightly closer to the DOT average. This adjustment compensates for the difference between DOT and DASH time duration variances. The resulting pulse decision boundary lies nearly in the center of the gap between DOT and DASH time duration clusters. If a new pulse has a time duration greater than the pulse average, it is considered to be a DASH; otherwise it is a DOT. Recomputation of the pulse average after receipt of each DOT and DASH permits the threshold

to adjust to slowly varying changes as they occur.

The DOT and DASH averages are influenced most by the DOT-SYMBOL and DASH-SYMBOL clusters shown on the data distribution plots. This is due to the fact that there are proportionately more pulses followed by SYMBOL spaces than followed by CHARACTER or WORD spaces. Since the DOT and DASH time duration variance is smaller in these clusters, especially in the DASH-SYMBOL cluster, the resulting pulse average, as determined by equation (3.2), lies more near the center of the gap between the two clusters, thereby providing a better linear decision boundary.

Space Algorithms. The wide variance of CHARACTER and WORD space time durations prohibits use of the averaging technique used for the pulse algorithms. The technique was tried, however, with less than desirable results. A threshold was established as the mid-point between the CHARACTER space average and the WORD space average. New spaces were classified as CHARACTER spaces if their time duration was less than the threshold, and as WORD spaces if their time duration was greater than the threshold. Respective space averages were then computed in a manner similar to that used in the pulse algorithm. In test runs of the recognition program, the threshold point was consistently smaller than the optimum value, resulting in many CHARACTER spaces being classified as WORD spaces.

This lower than desired threshold is due to two contributing factors. The first, and most important, is the large overlap of the CHARACTER and WORD space variances. The second factor is the difference in frequency of occurrence of the two types of spaces. CHARACTER spaces, for English language text, occur approximately 4 times as often as WORD spaces. Those CHARACTER space time durations which are slightly greater than the

threshold value force the WORD space average to be lower than it actually is. This, in turn, lowers the threshold value, which is defined as the mid-point of the CHARACTER and WORD space averages. The lower threshold causes more CHARACTER spaces to be defined as WORD spaces, thus lowering the WORD space average even more and compounding the problem. The threshold value settles near the true CHARACTER space average, much lower than desired.

The space algorithm finally arrived at for use in the recognition program is based on the average of all non-SYMBOL spaces which follow a DOT. The CHARACTER-WORD space average is calculated as follows:

$$\text{C-W SPACE AVG.} = \text{C-W SPACE AVG.} - \frac{\text{C-W SPACE AVG.}}{8} + \frac{\text{SPACE TIME}}{8} \quad (3.3)$$

Figures 3-2, 3-3, and 3-4 illustrate the linear decision boundaries, as determined by the pulse and space algorithms, for Recording Sessions 1, 2, and 3, respectively. Individual pulse-space clusters have been circled on these figures to indicate the size and shape of each cluster and the overlap regions between clusters. Table II lists the type of cluster identified by the letters shown on the three figures.

TABLE II	
Cluster Identification for Figures 3-2, 3-3, and 3-4	
<u>Area</u>	<u>Cluster Type</u>
A	DOT-WORD
B	DOT-CHARACTER
C	DOT-SYMBOL
D	DASH-WORD
E	DASH-CHARACTER
F	DASH-SYMBOL

DOT & DASH (ALL)

RECORDING SESSION 1

(See Table II for cluster identification)

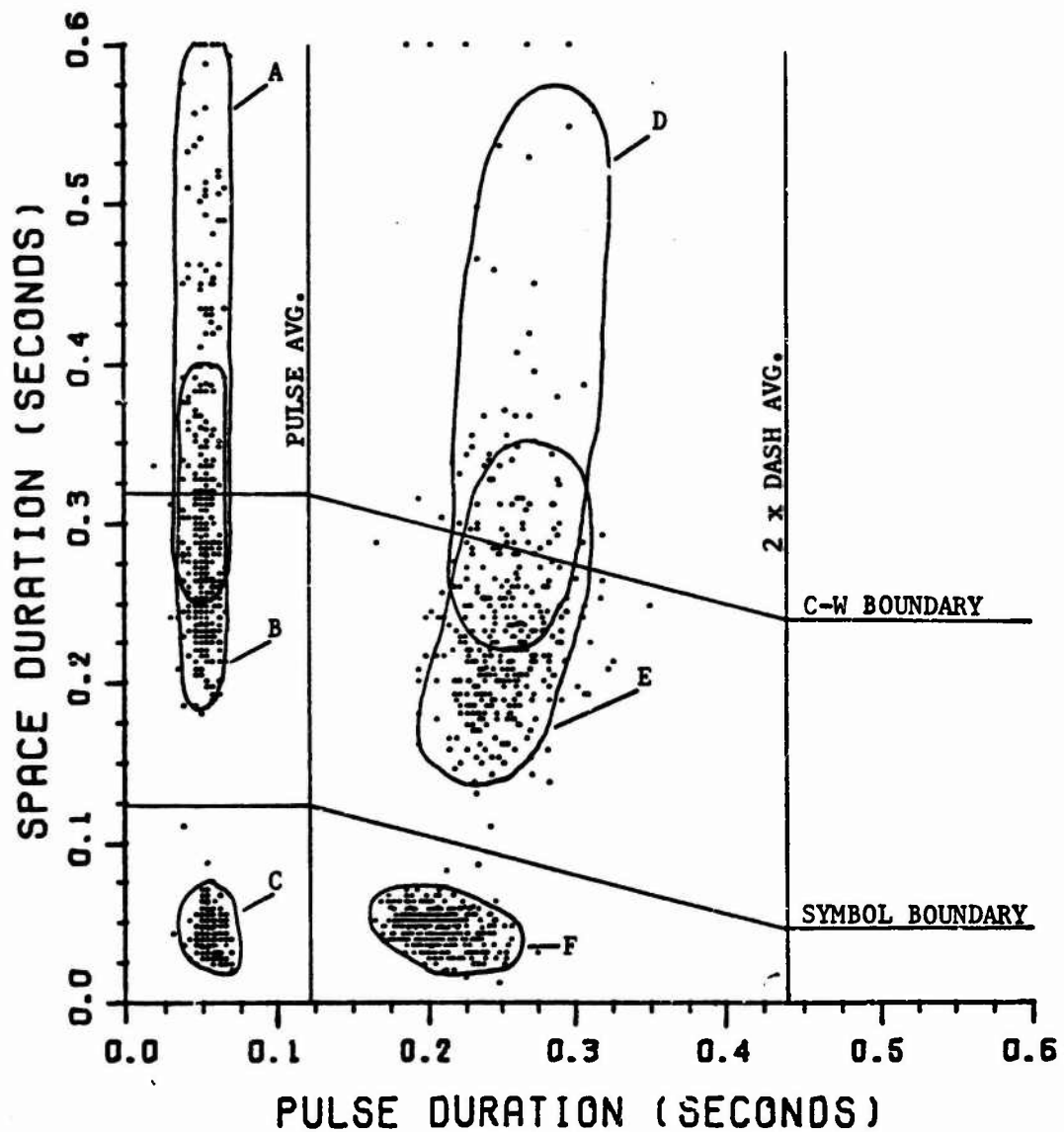


Fig. 3-2. Linear Decision Boundaries for Recording Session 1.

DOT & DASH (ALL)

RECORDING SESSION 2

(See Table II for cluster identification)

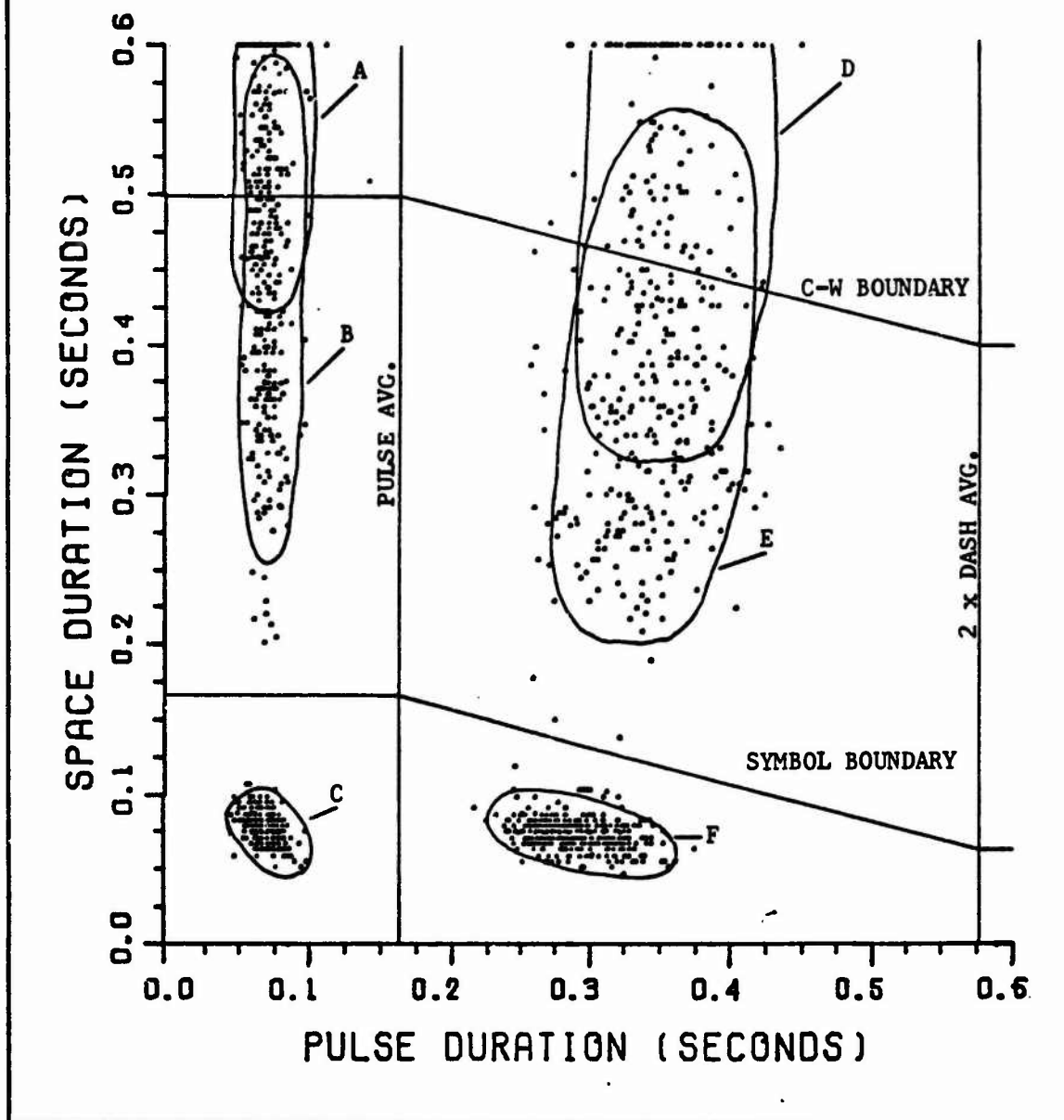


Fig. 3-3. Linear Decision Boundaries for Recording Session 2.

DOT & DASH (ALL)

RECORDING SESSION 3

(See Table II for Cluster identification)

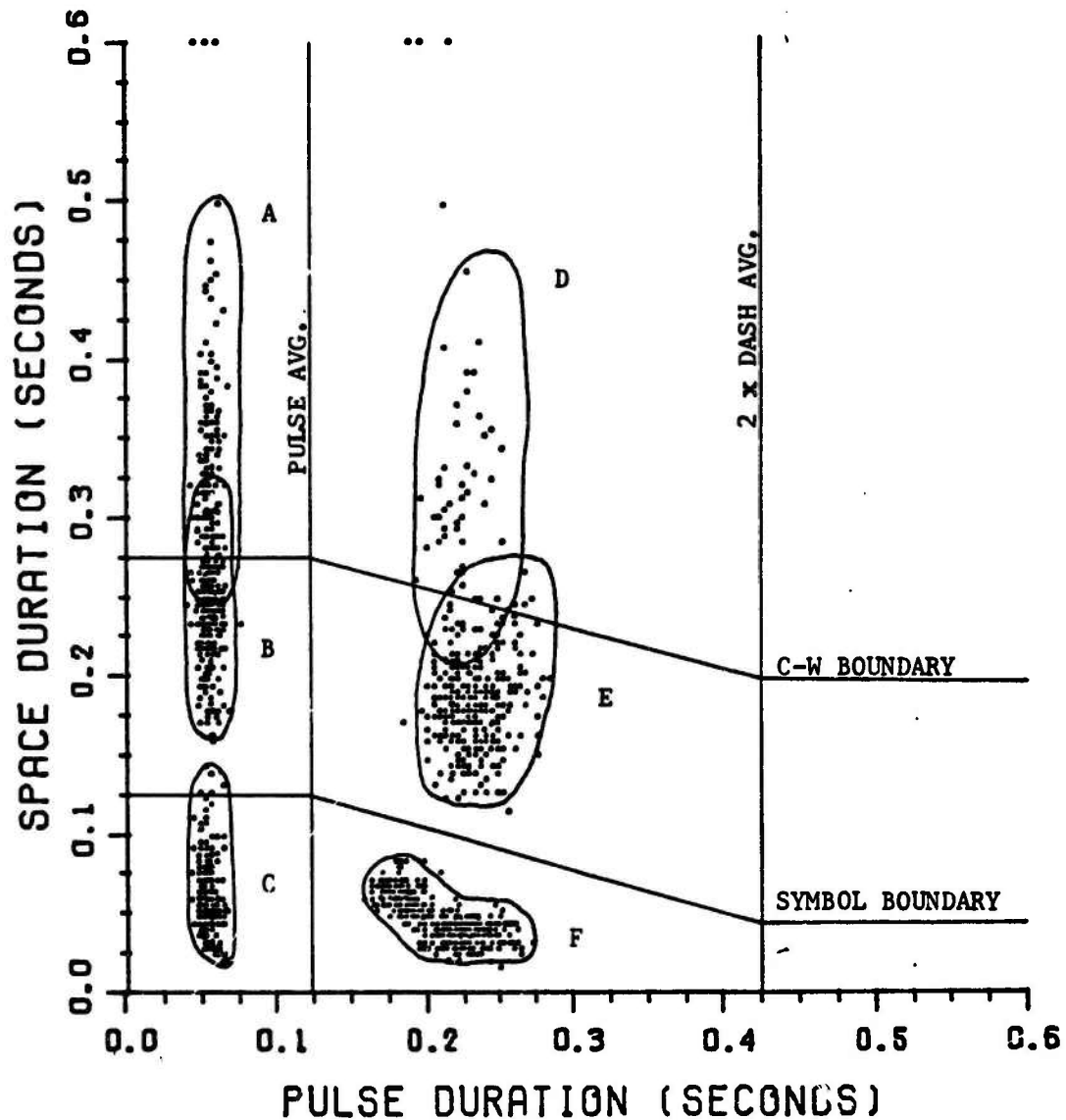


Fig. 3-4. Linear Decision Boundaries for Recording Session 3.

The pulse average lies in the gap between the DOT and DASH clusters on all three figures. This average value also lies in the gap between the DOT-SYMBOL cluster and the DOT-CHARACTER cluster for Recording Sessions 1 and 2, but not for Recording Session 3. Notice that the DOT-SYMBOL cluster in Fig. 3-4 has a much larger SYMBOL space variance than is indicated on similar clusters shown in Figures 3-2 and 3-3. This large variance is due to a faulty DOT-SYMBOL space generator on the "bug" used to transmit Recording Session 3. Therefore, it is assumed that the pulse average value also lies in the correct location for Recording Session 3. The pulse average is therefore used as the linear decision boundary for SYMBOL versus non-SYMBOL space decisions, when the space is preceded by a DOT. Also notice that the CHARACTER-WORD average, computed by equation (3.3), lies in the DOT-CHARACTER and DOT-WORD cluster overlap region.

CHARACTER and WORD spaces, when preceded by a DASH, have shorter time durations than those preceded by a DOT. Therefore, an adjusted boundary value must be used when a space is preceded by a DASH. The adjusted SYMBOL space boundary value is computed as a function of the time duration of the preceding DASH as follows:

$$\text{SYMBOL BOUNDARY} = \text{PULSE AVG.} - \frac{(\text{DASH TIME} - \text{PULSE AVG.})}{4} \quad (3.4)$$

The adjusted CHARACTER-WORD space boundary value is computed in a similar manner as follows:

$$\text{C-W BOUNDARY} = \text{C-W SPACE AVG.} - \frac{(\text{DASH TIME} - \text{C-W SPACE AVG.})}{4} \quad (3.5)$$

The resulting linear decision boundaries are negatively sloping lines, as shown on the accompanying figures. A maximum DASH time duration is established as twice the current DASH average for use in the boundary computations, thereby setting a lower limit to the threshold value.

To permit real-time adjustment of the C-W Boundary, a positive or negative adjustment value, X, is added to the threshold as follows:

$$\text{ADJUSTED C-W BOUNDARY} = \text{C-W BOUNDARY} \pm X \quad (3.6)$$

By adding the appropriate adjustment value to suit a particular Morse code transmission, the operator can optimize the readability of the printed output.

IV. The Computer Recognition Program

This Chapter describes the operation of the computer recognition program. The program consists of two distinct but interdependent parts: 1) the Signal Processor section, and 2) the Code Translation section. Discussion of the routines contained within each section is presented in an order indicative of the overall organization of the recognition program and the sequence in which the routines are executed. General flow charts are included for each routine to supplement the associated text.

A discussion of the programming constraints due to the physical limitations of the PDP-12 computer and the self-imposed operational goals is presented at the beginning of this chapter. It is hoped that knowledge of these limitations will enable the reader to better understand the rationale behind various operations performed within the program. A complete program listing with comments is provided in Appendix A.

Program Constraints

Before ideas can be constructively transformed into computer programs, the operational characteristics of the particular computer system to be used must be defined. A list of operational goals and performance objectives must also be defined to optimize the programming process.

PDP-12 Description. A brief overview of the characteristics of the DEC (Digital Equipment Corporation) PDP-12 (Programmed Data Processor-12) computer is provided in this section. A more complete description of the PDP-12 may be obtained in the PDP-12 System Reference Manual (Ref 2).

The PDP-12 is a versatile digital computer which contains two distinct

operating modes within its central processor, each with its own instruction set. The central processor logic is fully parallel, using a basic word length of 12 bits. The processor cycle time is 1.6 microseconds $\pm 20\%$. Most instructions require from 1 to 3 cycles for execution. The PDP-12 operates in one mode as a LINC (Laboratory Instrument Computer) and in the other mode as a PDP-8 computer. The computer may be stopped and started in either mode. Both operating modes have equal priority and programs may be switched from one to the other at will. Computations in one mode are immediately available to programs operating in the other mode because only one set of processing registers is involved.

The principal unit of core memory is a module of 4096 (4K) 12-bit words. Up to seven additional modules may be added, providing a total of 32,768 words. The logical organization within each module depends on the operating mode. In LINC mode, each 4K module is divided into four 1024-word segments. Only two of these segments are active at any given time: 1) the Instruction Field, which contains the executable program and directly addressed data, and 2) the Data Field, which contains only indirectly accessed data. In 8 mode, each 4K module (memory field) is divided into thirty-two 128-word pages. Data may be directly addressed to the current page or to page 0 only. Indirect addressing, through page 0, must be used to address data between pages. Special instructions must be used to change Instruction Field or Data Field segments in LINC mode and to change memory fields in 8 mode.

Many of the peripheral devices available with the PDP-12 are controllable only in LINC mode. Of these, the A-D Converter and the C.T. Display are used in the recognition program listed in Appendix A. The CRT Display, however, is not essential to the program and its use

will not be described in this report. The other peripheral devices used in the program, the Programmable Real-Time Clock and the Teletype device, may be controlled in either mode. A brief description of these devices and their use in the recognition program is given in the following paragraphs.

The A-D Converter consists of eight external input channels and eight internal input channels. The external input channels have an acceptable input voltage range of $\pm 1\text{V}$, corresponding to a sample value range of $\pm 777_8$. The internal input channels (control knobs) also have a sample value range of $\pm 777_8$. One external input channel is used in the recognition program to sample the voltage level of the Morse code signal. Three internal input channels are used to set the input signal threshold level, the number of samples to be averaged, and the CHARACTER-WORD boundary adjustment value.

The Programmable Real-Time Clock consists of a 400 kHz crystal clock, a 12-bit counter register, and an overflow bit. The clock may be used to synchronize the central processor to external events, count external events, measure intervals of time between events, or provide program interrupts at intervals from 2.5 microseconds to over 40 seconds. The 400 kHz crystal clock may be used to provide pulses to the counter register at 100 Hz, 1 kHz, 10 kHz, 100 kHz, or 400 kHz rates; or an external source may be used to drive the counter. The clock is used in the recognition program to measure the time durations of pulses and spaces. An external source is used to permit variable counter rates in the 1 kHz to 10 kHz range.

The Teletype device is used to type in or print out information at a rate of up to ten characters per second. Similar devices, such as the

DECWRITER, operate at much faster rates. A DECWRITER was used during this project to print the test messages shown in Appendix D.

Certain distinguishing features of the two operating modes must be considered when changing modes within a program. One of these is the addressing scheme. In LINC mode, Instruction Fields and Data Fields consist of 1024-word segments. The addresses within each field range from 0000_8 to 1777_8 , regardless of the physical location in core memory. Thus, location 0100_8 in Data Field 3 corresponds to physical location 6100_8 in core memory. In 8 mode, address locations correspond exactly to physical locations.

Another distinguishing feature between the two modes concerns arithmetic operations. LINC mode uses 1's complement addition for most operations, whereas 2's complement addition is used in 8 mode. As an example, 7777_8 is interpreted in LINC mode as -0 and in 8 mode as -1. Also, $7777_8 + 0001_8$ yields 0001_8 in LINC mode and 0000_8 in 8 mode.

Operational Goals. It goes without saying that the overall operational goal is accuracy. The output of the computer recognition program should exactly reproduce the hand-sent Morse code transmission. But by what measure should the accuracy of the program be evaluated? Should the program output be compared with that which a human would interpret as the transmitted message? Or should it be based strictly on the quality of the received code, as compared to the standards for Morse code language? Both evaluation criteria merit consideration. Further discussion of this topic is presented in Chapter VI.

Three operational goals are specified for the computer recognition program. These are:

1. The program must be able to process the received Morse code transmission in real-time.
2. The final program must occupy less than 4K of memory space.
3. The program should be designed to lend itself to construction of a small, low-cost hardware realization. i.e., a special purpose minicomputer.

These goals are somewhat interrelated. If the recognition program cannot process the received Morse code in real-time, a memory storage unit is required to save the received signal until it can be processed, thus requiring additional core space. In this case, the size of the memory will determine the maximum length message that could be processed at any one time. The real-time constraint eliminates this problem.

The 4K memory limitation was chosen to permit implementation of the program within one basic unit of memory. Thus, the recognition program can be implemented on the basic PDP-12 or similar minicomputer.

Construction of a hardware realization of the recognition program is of particular interest at the Air Force Institute of Technology. Aside from obvious educational benefits, construction of a low cost, portable Morse code recognition machine has many military and civilian applications. Proper choice of computer instructions, along with compliance to the aforementioned goals, will result in a minimum number of components necessary to construct a hardware realization.

Program Description

The recognition program is segmented into two major parts: the Signal Processing section (Fig. 4-1), and the Code Translation section

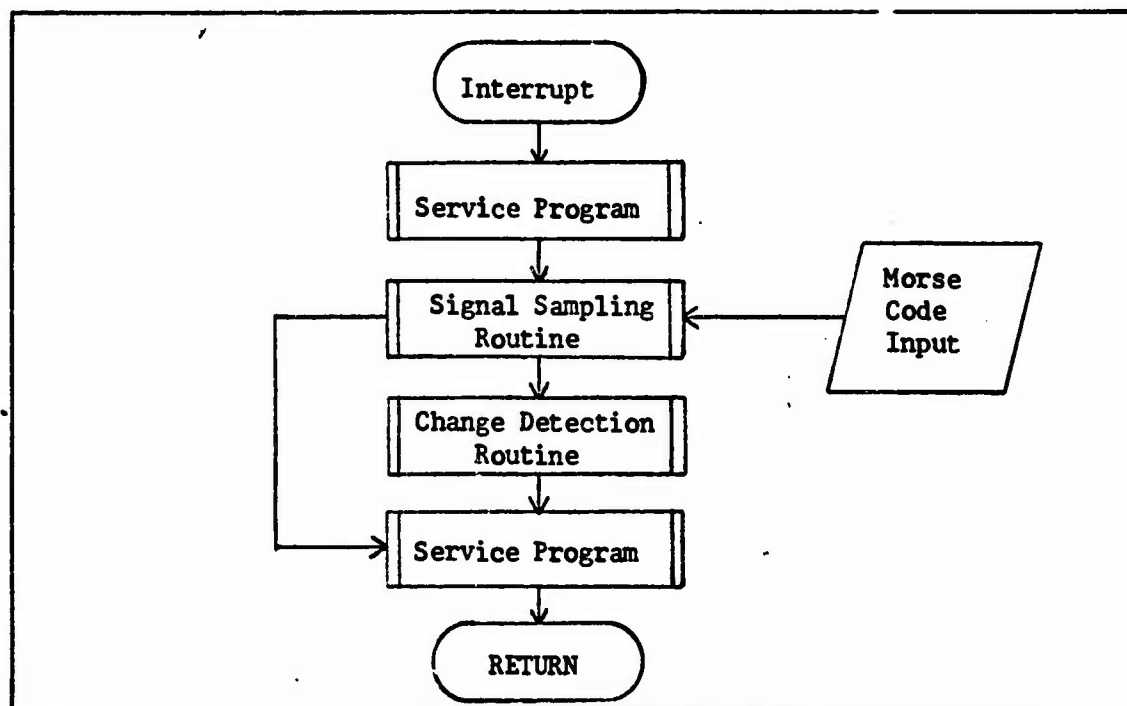


Fig. 4-1. Signal Processing Section Organization.

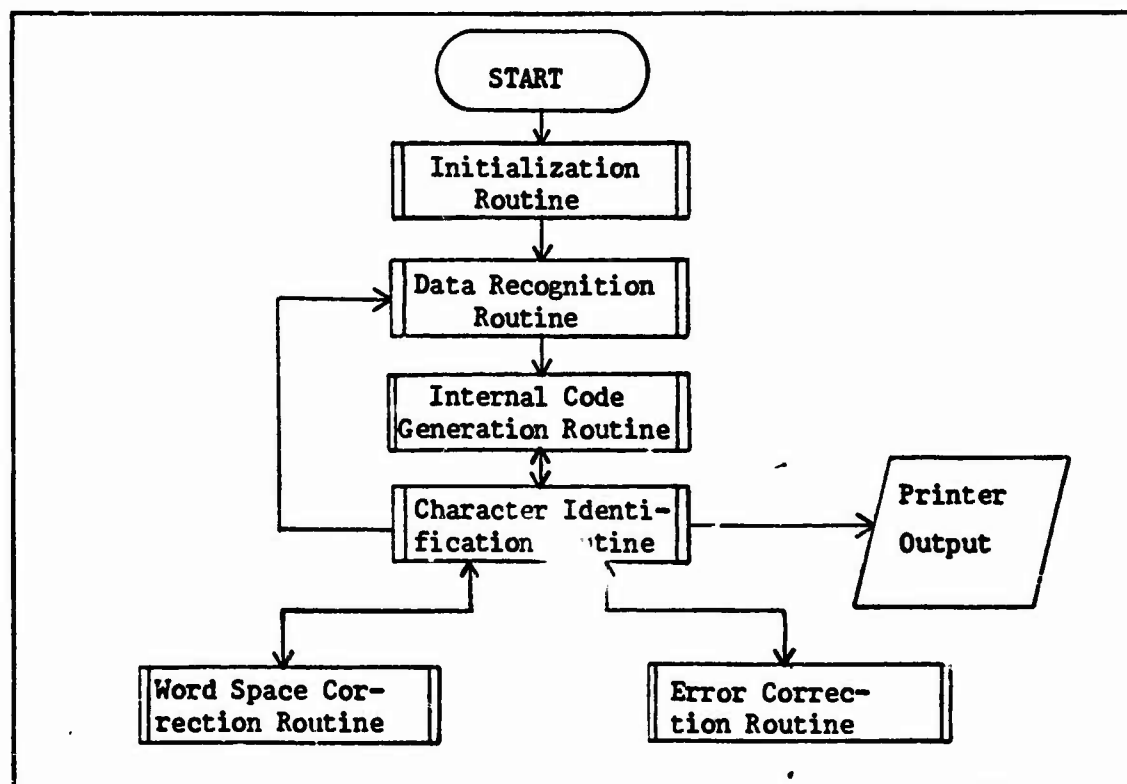


Fig. 4-2. Code Translation Section Organization.

(Fig. 4-2). The Signal Processing section serves to transform the received Morse code analog signal into pulse and space time duration information for use in the Code Translation section. The Code Translation section then uses these time durations to identify transmitted Morse code characters and print out the message. Detailed discussion of these two sections and their associated routines is presented in subsequent paragraphs.

Program operation in either section is controlled by an external program interrupt. Acknowledgement of a program interrupt is controlled by the status of an interrupt bus. If the bus is enabled, a program interrupt will halt current computer execution and transfer operation to absolute memory location 0001_8 in 8 mode or 0040_8 in LINC mode. If the interrupt bus is disabled, a program interrupt has no effect on execution. When a program interrupt is recognized and operation is transferred to the appropriate location, the interrupt bus is automatically disabled. The interrupt bus can only be enabled again by specific program instruction.

The interrupt bus is enabled during operation in the Code Translation section only. When an interrupt occurs, execution in that section is halted and the Service program (Fig. 4-3) is entered through location 0001_8 or 0040_8 , depending on the computer mode at the time of interrupt. In either case, the current accumulator, link, and program counter register values are saved for use in returning to interrupted program. Signal Processing section operation then begins and continues until complete, at which time the Service program is reentered. Appropriate register values are restored, the interrupt bus is turned on, and operation again resumes in the Code Translation section at the point of interruption. Operation continues in this section until the occurrence of another inter-

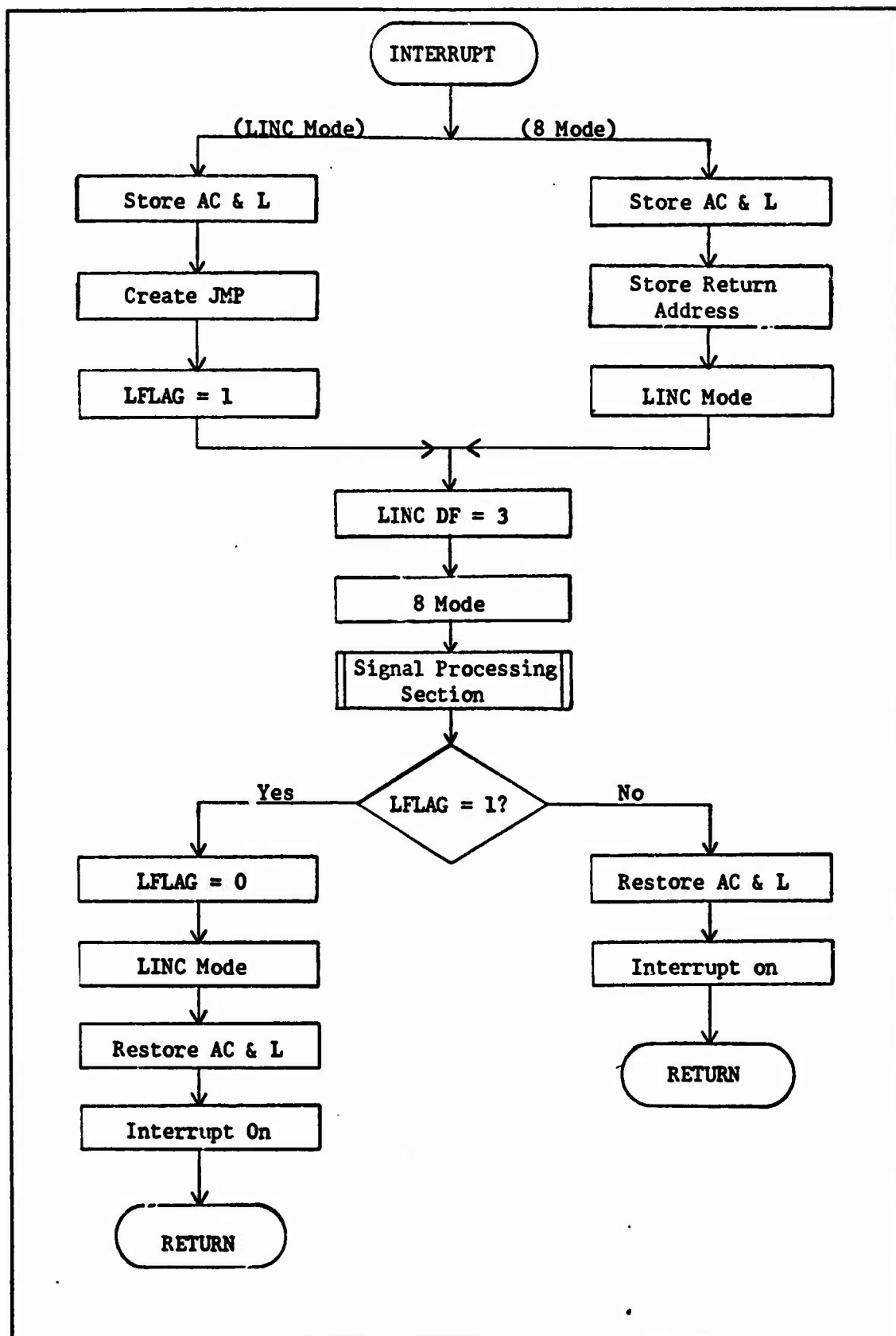


Fig. 4-3. Service Program Flowchart.

rupt, at which time the process is repeated.

Signal Processing Section

The Signal Processing section, as previously mentioned, serves to transform the received analog Morse code signal into pulse and space time duration information for use in the Code Translation section. If the received signal were completely clean and interference free, this transformation would be relatively easy. However, the transformation becomes rather difficult when the effects of noise and signal fading are considered. The main purpose, then, of the Signal Processing section is to recognize the Morse code transmission under realistic conditions of signal fading and distinguish it from pulse interference and "white" noise. The methods used to achieve these tasks will now be discussed.

The incoming pulse-modulated signal is first converted into dc pulses, corresponding to the Morse code being received, by a full-wave bridge rectifier and RC filter. The input signal voltage level is sampled periodically (approximately once every 200 microseconds) and compared to a threshold voltage level. If the input signal is less than the threshold level, a -1 is stored; if the input signal is greater than the threshold level, a +1 is stored. The effects of signal fading may be reduced by proper settings of the input signal voltage and threshold level.

A low pass digital filter technique is employed to limit the effects of noise on the input signal. Rather than base the pulse-space decision on whether the input signal level is above or below the threshold level for any one particular sample, the average of several of the samples must be above or below the threshold before a pulse-to-space or space-to-pulse

change is detected.

When an input signal level change is detected, it is checked again to determine if the change is permanent. The checking process is similar to the change detection process, except that twice as many samples are averaged. If the checking process result confirms the input signal change, then the signal is considered to have changed permanently. The time duration of the just-ended pulse or space is then obtained from the real-time clock counter and the clock is reset to begin timing the next pulse or space. If the result of the checking process conflicts with the change detection process, then the input signal change was due to interference rather than a valid Morse code input. In this case, the change detection process is repeated, again looking for an input signal level change.

When a valid input signal change is detected, the pulse or space time duration is stored in a 200₈-word memory buffer. Previously stored time durations are retrieved from the buffer as needed by the Code Translation section for processing. The 200₈-word buffer permits the Code Translation section to temporarily lag behind the Signal Processing section without loss of Morse code signal information.

This method of storage and retrieval of Morse code time durations presents an operational limit to the overall recognition program in that it is possible to over-write previously stored time durations before the Code Translation section processes them. If the interrupts occur too frequently, not enough time will be available for the Code Translation section to keep up with the storage of new time durations, a function of the Morse code transmission speed. Thus, the interrupt frequency upper limit is determined by the execution time of the Code Translation section. Fortunately, this presents no real problem to the overall recognition

program, since the upper limit is well above the minimum input signal sample frequency needed to ensure proper detection of Morse code signal changes.

The Signal Processing section is divided into two separate routines: 1) the Signal Sampling routine (Fig. 4-4), and 2) the Change Detection routine (Fig. 4-5). The particular function of these routines is discussed in the following paragraphs.

Signal Sampling Routine. The Signal Sampling routine performs the actual sampling of the Morse code input signal. If the sampled input voltage is greater than the threshold level, a +1 is stored. If the sampled input voltage is less than the threshold level, a -1 is stored. If N' samples have not been taken yet, program operation is returned to the Code Translation section through the Service program. When the N'th sample is taken, program operation continues in the Change Detection routine.

Change Detection Routine. The Change Detection routine performs the change detection and checking processes. After the N'th signal sample has been taken, the N' +1s are added together to yield a positive or negative result. A +1 is assigned for a positive result and a -1 is assigned for a negative result. This value is then compared with the + or - 1 representing the current input signal level. If the comparison indicates that a change has occurred, the checking process is initiated to confirm or deny the change. If the comparison indicates that no change has occurred, program operation is returned to the Code Translation section through the Service program and the N' sampling process is started again.

As an example, assume that a Morse code pulse is being received.

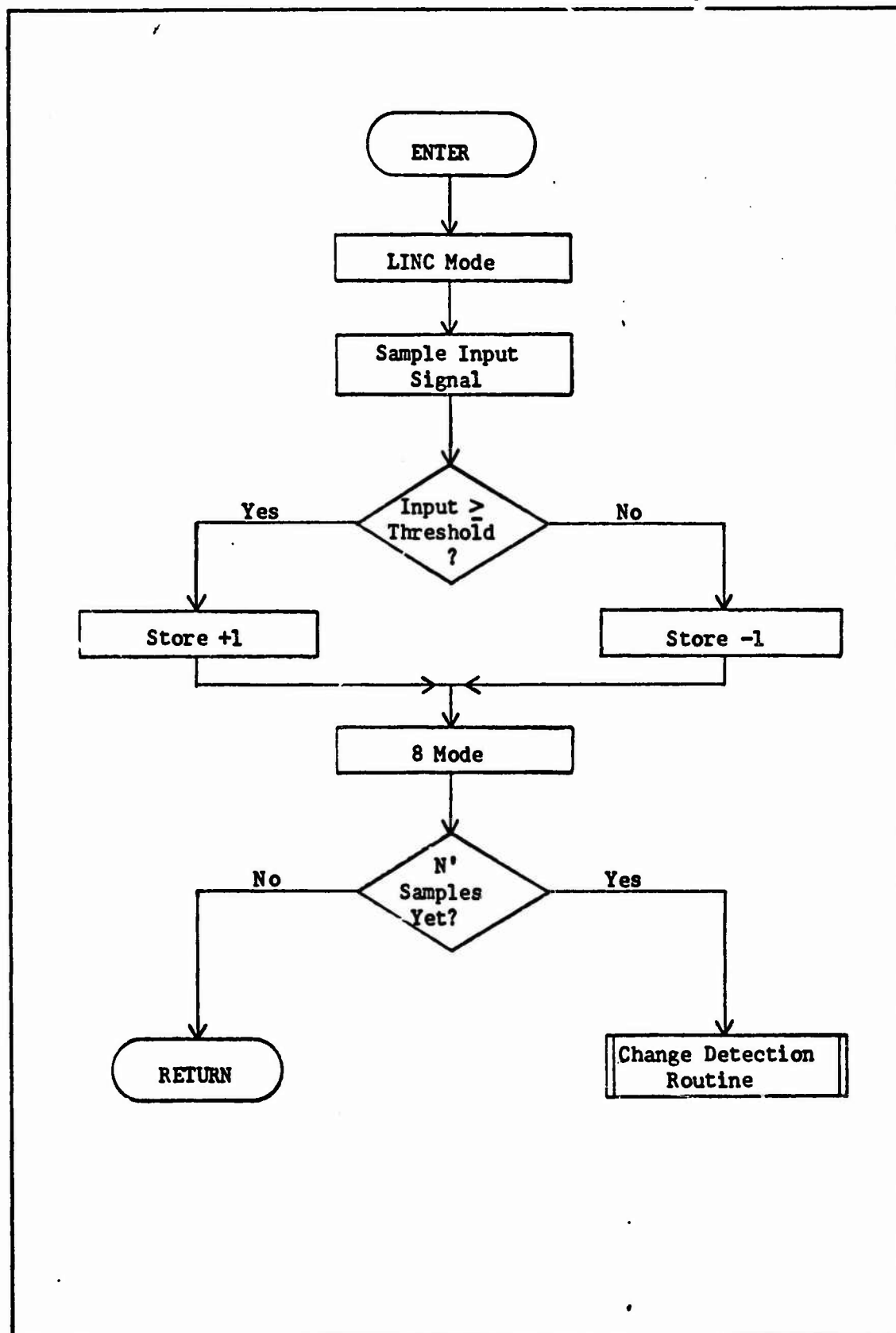


Fig. 4-4. Signal Sampling Routine Flowchart.

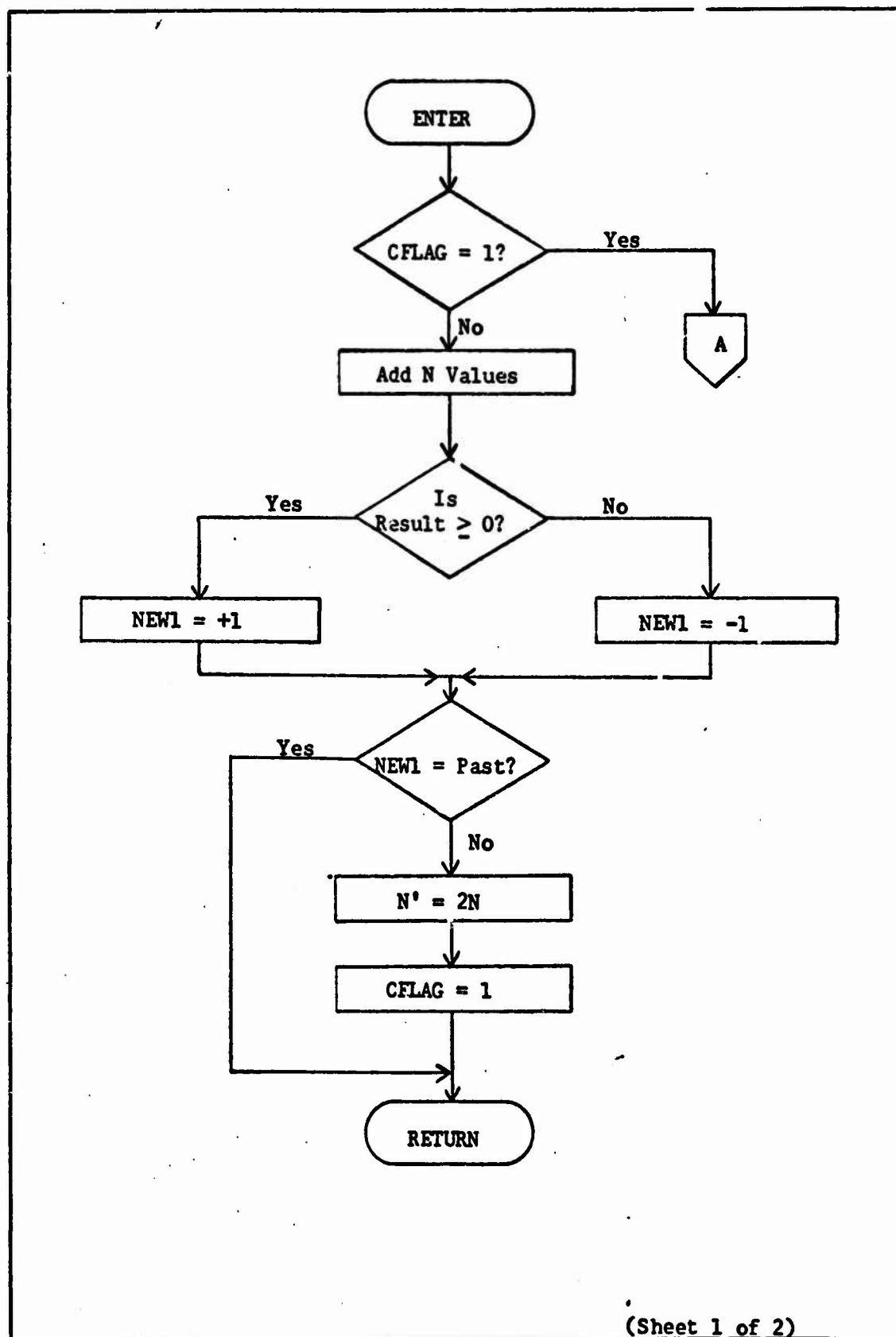


Fig. 4-5. Change Detection Routine Flowchart

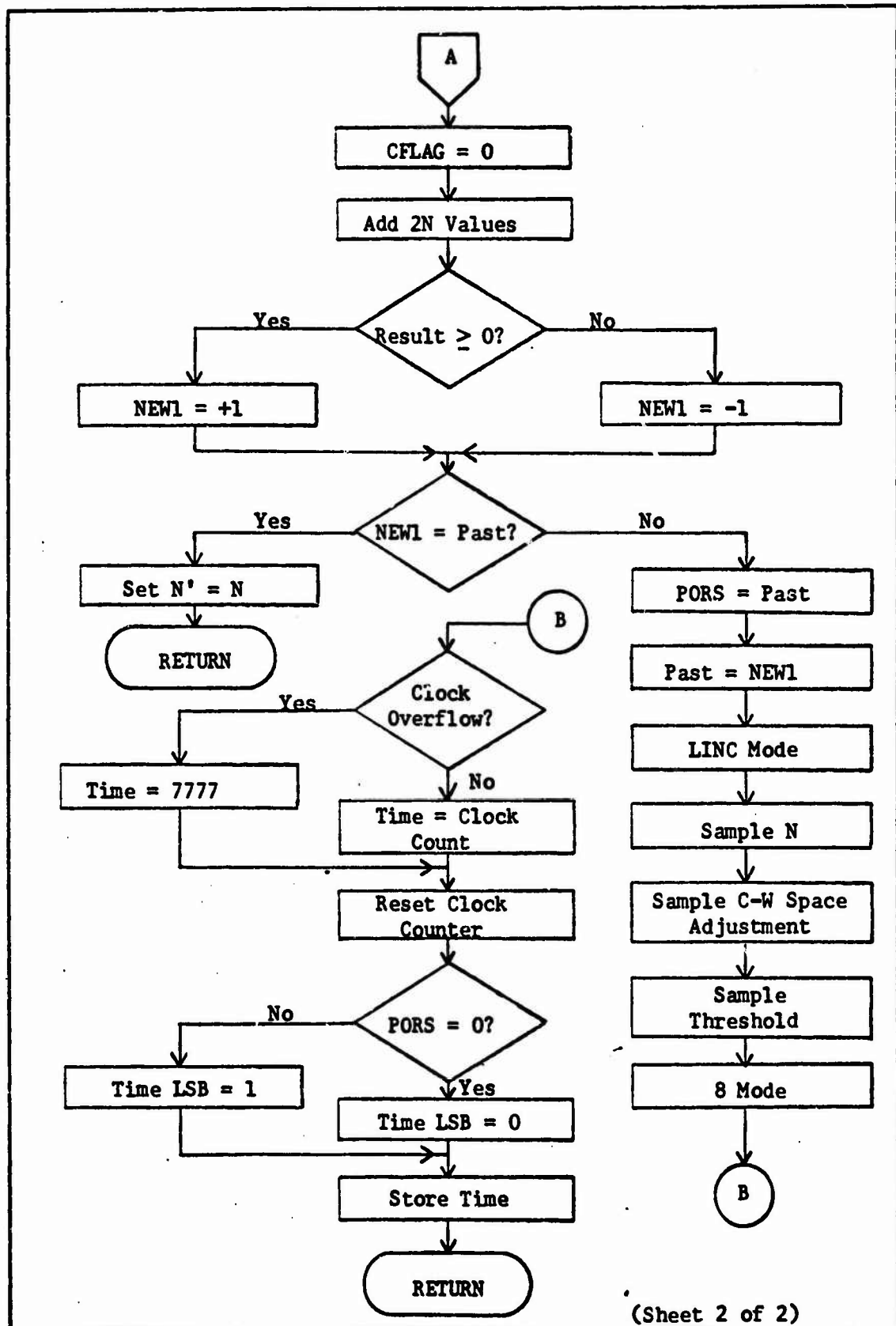


Fig. 4-5. Change Detection Routine Flowchart.

When the input signal changed from the previous space to the current pulse, the time duration of the space was recorded and stored, and a +1 was stored to indicate that a pulse is currently being received. After every N' samples are averaged, the +1 is compared with the stored +1 value. A change is indicated when the N' sample average is -1. The checking process then begins. If the result of the checking process also indicates a -1, then the input signal is considered to have permanently changed to the succeeding space and the time duration of the just completed pulse is read from the real-time clock and stored. This time duration is then marked as a pulse by setting the least significant bit (LSB) of the time duration to a 1. Likewise, the LSB is set to a 0 for a space. In this way, the Code Translation section can correctly identify pulses and spaces.

The real-time clock counter has a range of 0000_8 to 7777_8 . If a space or a pulse is long enough, the counter will count past 7777_8 and restart at 0000_8 . When this occurs, an overflow bit is set. Subsequent overflows will not reset the overflow bit. When a permanent change is detected the overflow bit is checked first before the clock counter is read. If an overflow has occurred, a clock value of 7777_8 is automatically assumed. The actual clock counter value is read only if an overflow has not occurred, thereby preventing erroneous time durations from being used in the program.

The actual time duration of a pulse or space is determined by the real-time clock frequency. This frequency is set to permit counter readings for the longest pulse or space to be below the overflow condition. Thus, overflows will normally only occur when there is a long pause between Morse code transmissions.

Code Translation Section

The Code Translation section converts the stored pulse and space time duration information produced by the Signal Processing section into a printed copy of the received Morse code message. This section is composed of six routines: 1) the Initialization routine, 2) the Data Recognition routine, 3) the Internal Code Generation routine, 4) the Character Identification routine, 5) the Word Space Correction routine, and 6) the Error Correction routine.

Recognition program operation begins in the Code Translation section. Pulses and spaces are identified by comparing the time duration of a particular pulse or space to the average of past pulses and spaces. Before any decisions can be made in this manner, some a priori knowledge of the particular averages must be obtained. The Initialization routine provides this knowledge by examining the first 49 pulses received, in a two-step process. The acquired knowledge is then used to start the recognition process, commencing with the first pulse or space received. As subsequent pulses and spaces are processed, the averaging information is constantly updated to adjust to changes in Morse code sending rates.

Pulses retrieved from the 200₈-word memory buffer are compared with the pulse average. If the time duration of the new pulse is greater than this average, it is classified as a DASH, otherwise it is a DOT. The receipt of a DASH is noted by storing a 1 in a word register. A 0 is stored in the word register to indicate the receipt of a DOT. The number of pulses received is recorded by incrementing a number register.

Spaces are classified in a two-step process. First, the time duration of the space in question is compared with the SYMBOL space boundary value. If the time duration is less than this boundary value, the new space is

classified as a SYMBOL space. If the time duration is greater than the boundary value, the new space is either a CHARACTER or a WORD space. The second step of the space classification process is then used to make this distinction. The time duration of the new space is compared with the CHARACTER-WORD space boundary value. If the time duration is less than this value, the new space is classified as a CHARACTER space; otherwise it is classified as a WORD space.

When either a CHARACTER space or a WORD space is identified, the contents of the word and number registers are combined to yield a unique internal code word representing the received Morse code character. This internal code word is then compared with a list of internal code words stored in memory. The corresponding ASCII Teletype code is identified and used to print the character. A list of internal code words is presented in Appendix C.

When the internal code word cannot be identified as a valid code word, an error correction process is initiated. This process, when possible, corrects two types of errors: 1) the inclusion of an extra pulse in the code word due to noise in the received Morse code signal, and 2) the joining of two Morse code characters caused by too small a space separating the characters. In the first case, pulses having a time duration less than one-half of the current DOT average are eliminated. The internal code word is then recomputed and the valid character is identified. If no pulses can be eliminated in this process, the second case is considered. The largest SYMBOL space in the invalid code word is reclassified as a CHARACTER space. Two new internal code words are generated from the invalid code word and reprocessed. If either of the corrected internal code words is still invalid, a special error symbol is

printed to indicate receipt of an unidentifiable Morse code character.

Recognition of a WORD space causes the Teletypewriter to skip a space, thereby forming words rather than an endless string of transmitted characters. However, the overlap of CHARACTER and WORD space time durations, as shown in Chapter III, causes some CHARACTER spaces to be erroneously classified as WORD spaces. In order to limit the number of erroneous classifications, a separate routine is used to re-evaluate WORD spaces following the letters I, J, Q, U, V, and Z. The occurrence of these letters as the last letter of an English language word is highly unlikely, although not impossible. This routine compares the time duration of the space in question with a slightly larger boundary value. If the time duration is greater than this new boundary value, the space is classified as a CHARACTER space. The readability of the printed message is greatly improved through the use of this technique.

Discussion of the particular functions performed by each of the six routines contained in the Code Translation section is now presented in the following paragraphs.

Initialization Routine. The Initialization routine (Fig. 4-6) is entered from the Data Recognition routine. Operation begins by enabling the interrupt bus. Then the first 49 pulses (98 stored pulse and space time durations) are examined to establish initial average information. This process is accomplished in two steps. First, the time durations of the first 16 pulses are averaged to establish approximate DOT, DASH, and pulse averages according to the following set of equations:

$$\text{DOT AVG.} = \frac{\text{DOT AVG.}}{2} + \frac{\text{NEW DOT}}{2} \quad (4.1)$$

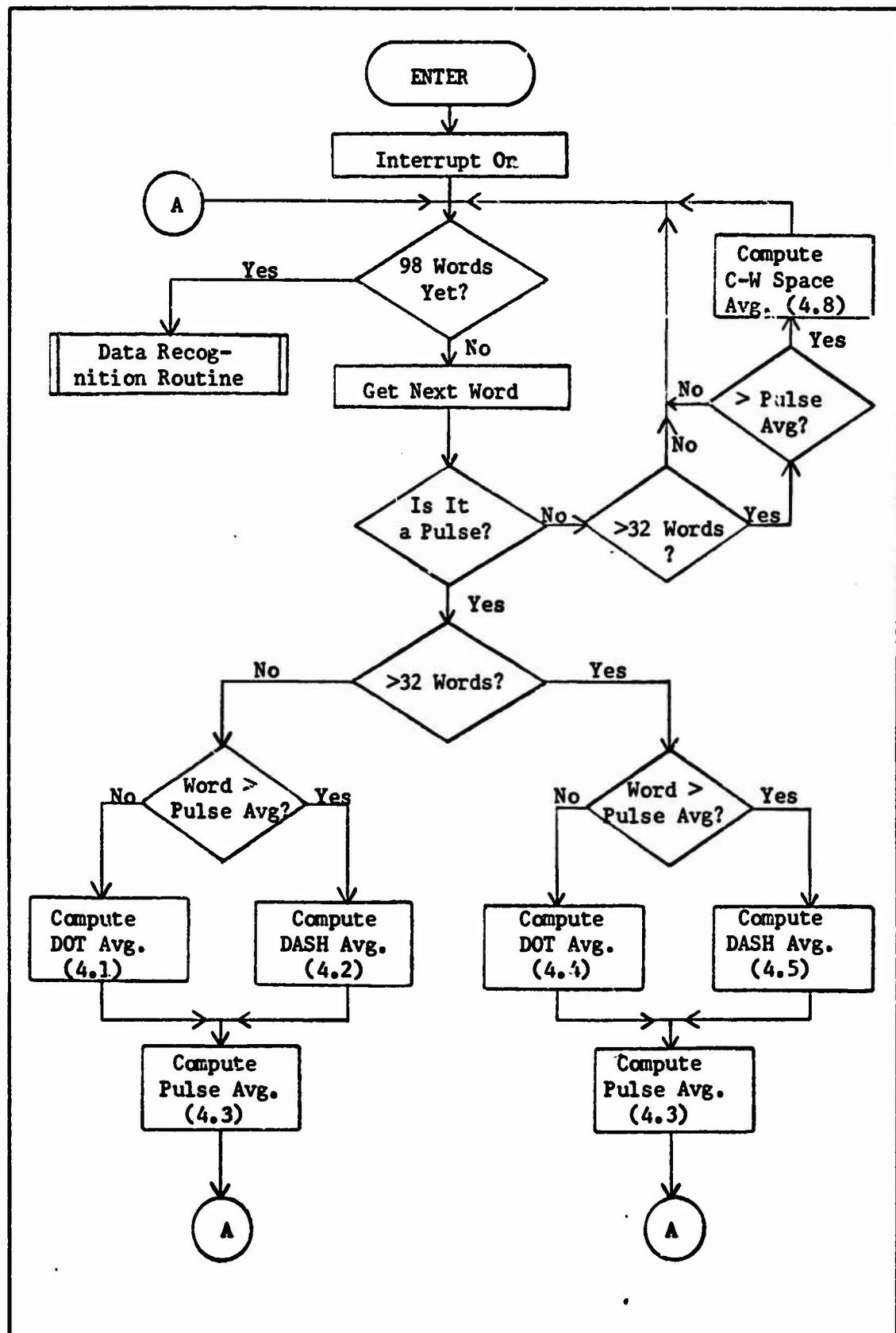


Fig. 4-6. Initialization Routine Flowchart.

$$\text{DASH AVG.} = \frac{\text{DASH AVG.}}{2} + \frac{\text{NEW DASH}}{2} \quad (4.2)$$

$$\text{PULSE AVG.} = \frac{\text{DOT AVG.}}{4} + \frac{\text{DASH AVG.}}{2} \quad (4.3)$$

All averages are initially set at 0. The first pulse is automatically classified as a DASH, regardless of its actual classification, because it is greater than the pulse average. The DASH and pulse averages are then computed according to the above equations. The second pulse is then compared with the pulse average and classified as a DOT or a DASH and the DOT average or DASH average is computed, as applicable. After three or four of each type of pulse has been processed, the resulting averages approach their true values. Note that the DOT and DASH averages are heavily influenced by the time duration of each new pulse. This permits the rapid establishment of initial averages; however, these averages are too sensitive to extreme deviations from the mean and are not desirable for long term use. Also note that the pulse average is not the mean of the DOT and DASH averages, but is, instead, shifted slightly towards the DOT average. As was shown in Chapter III, this slight compensation positions the pulse linear decision boundary in the center of the gap between DOT and DASH time duration clusters. Equation (4.3) is used throughout the recognition program to compute the pulse average.

In the second step of the Initialization routine, an improved set of DOT and DASH average equations, less sensitive to temporary time duration deviations from the mean, are used to refine the DOT, DASH and pulse averages:

$$\text{DOT AVG.} = \text{DOT AVG.} - \frac{\text{DOT AVG.}}{4} + \frac{\text{NEW DOT}}{4} \quad (4.4)$$

$$\text{DASH AVG.} = \text{DASH AVG.} - \frac{\text{DASH AVG.}}{4} + \frac{\text{NEW DASH}}{4} \quad (4.5)$$

The time durations of the remaining 33 pulses are averaged according to these equations. The pulse average, as always, is recomputed after each DOT or DASH average computation.

Upon completion of the initialization process, the DOT and DASH averaging equations are again changed to the following for use throughout the rest of the program:

$$\text{DOT AVG.} = \text{DOT AVG.} - \frac{\text{DOT AVG.}}{8} + \frac{\text{NEW DOT}}{8} \quad (4.6)$$

$$\text{DASH AVG.} = \text{DASH AVG.} - \frac{\text{DASH AVG.}}{8} + \frac{\text{NEW DASH}}{8} \quad (4.7)$$

The initial CHARACTER-WORD space average is also established during the Initialization routine. After the first 16 pulses have been examined and the preliminary pulse averages determined, the space averaging process begins. The space time duration is first compared with the pulse average to determine whether it is a SYMBOL space or not. If it is not, the CHARACTER-WORD space average is computed according to the following equation:

$$\text{C-W SPACE AVG.} = \text{C-W SPACE AVG.} - \frac{\text{C-W SPACE AVG.}}{4} + \frac{\text{NEW SPACE}}{4} \quad (4.8)$$

Upon completion of the initialization process, the CHARACTER-WORD

space average computation is performed only for non-SYMBOL spaces preceded by a DOT according to the following equation:

$$\text{C-W SPACE AVG.} = \text{C-W SPACE AVG.} - \frac{\text{C-W SPACE AVG.}}{8} + \frac{\text{NEW SPACE}}{8} \quad (4.9)$$

These final averaging equations allow the program to adjust to gradual changes without overly reacting to temporary fluctuations in pulse and space durations.

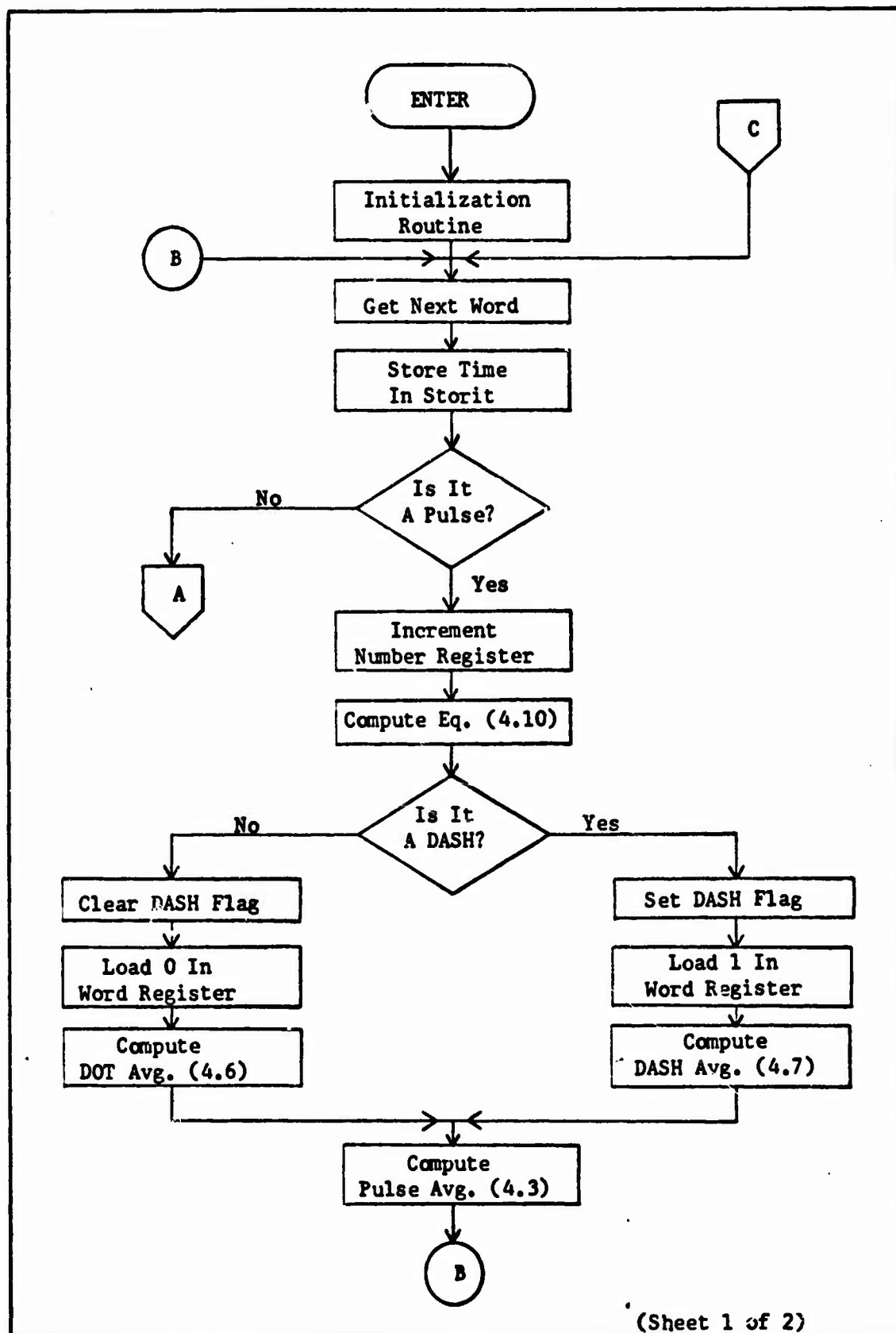
Data Recognition Routine. The Data Recognition routine (Fig. 4-7) begins after the initializing is complete. The first word is taken from the 200₈-word storage loop and identified as a pulse or space. If it is a pulse, it is compared to the pulse average and identified as a DOT or DASH by the following equation:

$$X = \frac{\text{PULSE AVG.}}{2} - \frac{\text{NEW PULSE}}{2} \quad (4.10)$$

$$(X \geq 0 \implies \text{DOT}; X < 0 \implies \text{DASH})$$

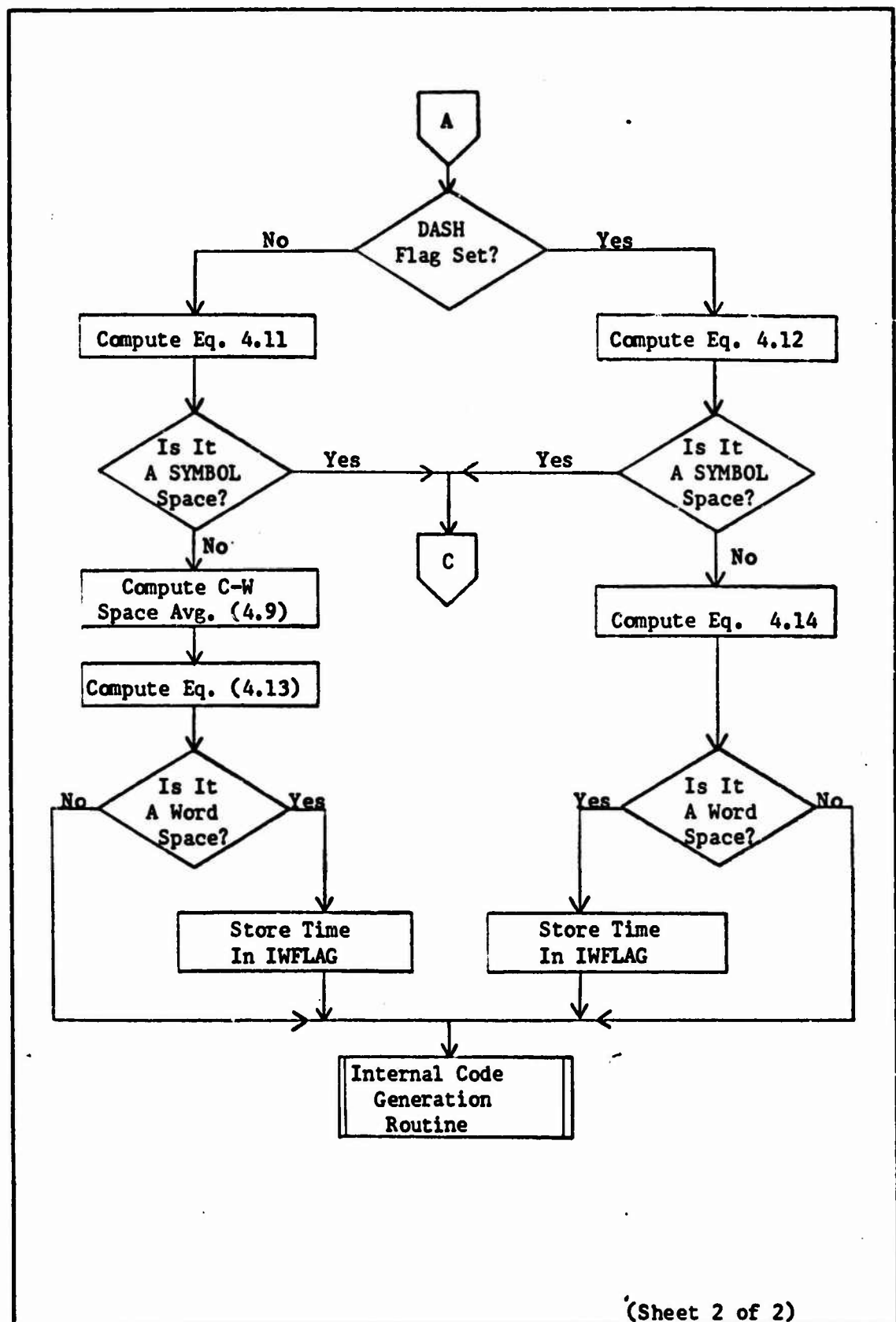
The respective averages are then recomputed to include the new input. A 0 or a 1 is stored in the word register to indicate receipt of the DOT or DASH, respectively, and the number register is incremented to keep track of the number of pulses received. The time duration of the pulse or space is successively stored in a 30₈-word memory location for use by the Error Correction process.

When a space is received, it is first categorized by whether or not



(Sheet 1 of 2)

Fig. 4-7. Data Recognition Routine Flowchart.



(Sheet 2 of 2)

Fig. 4-7. Data Recognition Routine Flowchart.

it follows a DASH. This distinction is necessary to permit use of the proper decision algorithms as specified in Chapter III. The next step is to determine if it is a SYMBOL space or not. When the space follows a DOT, the following equation applies:

$$X = \frac{\text{NEW SPACE}}{2} - \frac{\text{PULSE AVG.}}{2} \quad (4.11)$$

$(X < 0 \Rightarrow \text{SYMBOL SPACE})$

When the space follows a DASH, the equation is:

$$X = \text{NEW SPACE} - \frac{\text{PULSE AVG.} - \left[\frac{\text{DASH} - \text{PULSE AVG.}}{4} \right]}{2} \quad (4.12)$$

$(X < 0 \Rightarrow \text{SYMBOL SPACE})$

If the space is identified as a SYMBOL space, the Data Recognition routine is repeated by examining the next word. If the space is not a SYMBOL space, then it must be identified as either a CHARACTER space or a WORD space. Again, this decision depends on the type of preceding pulse.

When the space follows a DOT, the following equation is used:

$$X = \frac{\text{NEW SPACE}}{2} - \frac{\text{C-W SPACE AVG.}}{2} \quad (4.13)$$

$(X < 0 \Rightarrow \text{CHARACTER SPACE}; X \geq 0 \Rightarrow \text{WORD SPACE})$

When the space follows a DASH, the equation is:

$$X = \frac{\text{NEW SPACE}}{2} - \frac{\text{C-W SPACE AVG.} - \left[\frac{\text{DASH} - \text{PULSE AVG.}}{4} \right]}{2} \quad (4.14)$$

$(X < 0 \implies \text{CHARACTER SPACE}; X \geq 0 \implies \text{WORD SPACE})$

When the space is identified as a WORD space, the time duration is stored in a special register (IWFLAG) for use by the WORD Space Correction routine. Normally, the identification of a WORD space causes the Teletypewriter to skip a space following the printed character.

Internal Code Generation Routine. The identification of either a CHARACTER space or a WORD space signals the end of a Morse code character. The Internal Code Generation routine (Fig. 4-8) then forms the internal code word by combining the contents of the word register with the number register to yield a unique 12-bit code word (See Appendix C for a listing of internal code words).

An example of this process is shown in Fig. 4-9. A "P" (DOT-DASH-DASH-DOT) is received and correctly identified by the Data Recognition Process. As each pulse is received, a 0 or a 1 is stored in the least significant bit of the word register and the number register is incremented by 1. Previously received 0's or 1's stored in the word register are shifted left one position to allow room for the new pulse. When a CHARACTER or WORD space is received, the content of the word register is shifted left 12-N positions ($N = \text{number register value}$), the content of the number register is added, and the result is stored in the code register.

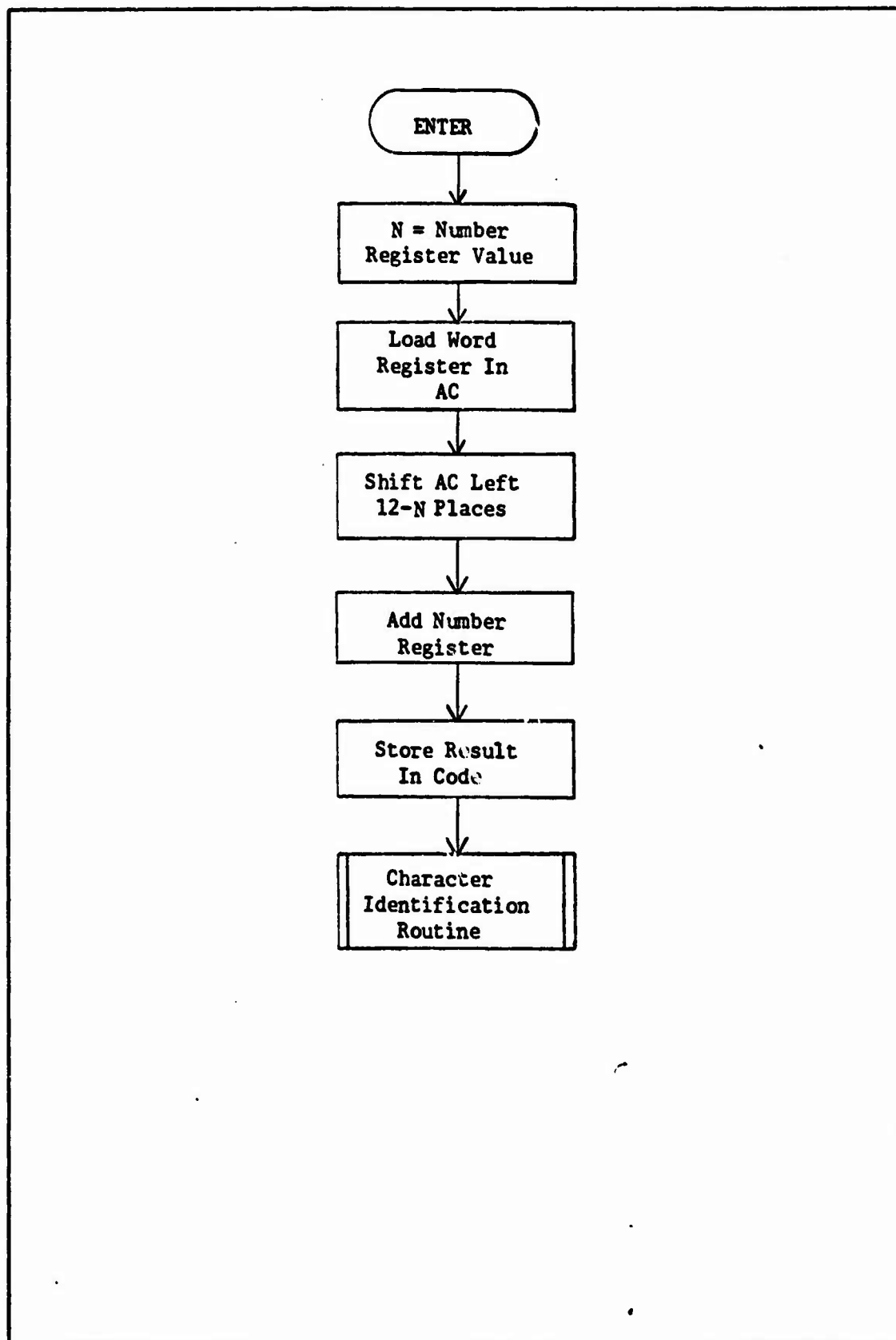


Fig. 4-8. Internal Code Generation Routine Flowchart.

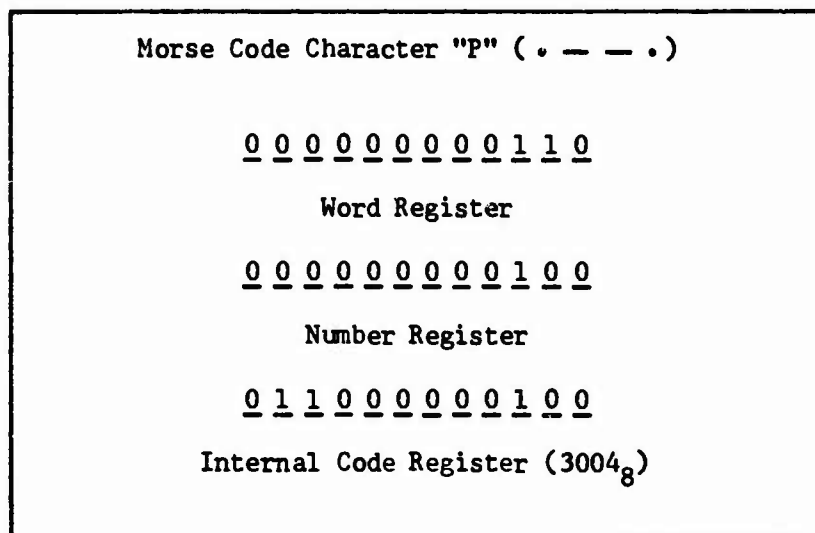


Fig. 4-9. Internal Code Word Generation

Because of the 12-bit register limitation on the PDP-12 computer, this internal process can only recognize Morse Code characters having eight or less pulses. Since most Morse code characters are 6 or less pulses long (error character has 8 pulses), this process is sufficient to handle the Morse code character set. The 12-bit register presents a limitation, however, to the Error Correction process, as discussed later in this chapter.

Character Identification Routine. The Character Identification routine (Fig. 4-10) compares the internal code word with a stored alphabet of 49 internal codes and selects the correct ASCII code for printer output. The comparison process has been divided into four steps to reduce computer execution time.

The first step determines which 4000₈ code subgroup (0000₈ → 3777₈ or 4000₈ → 7777₈) the internal code word falls in. The second step identifies the 1000₈ subgroup containing the internal code. The third step identifies which half of the 1000₈ subgroup the code word is in.

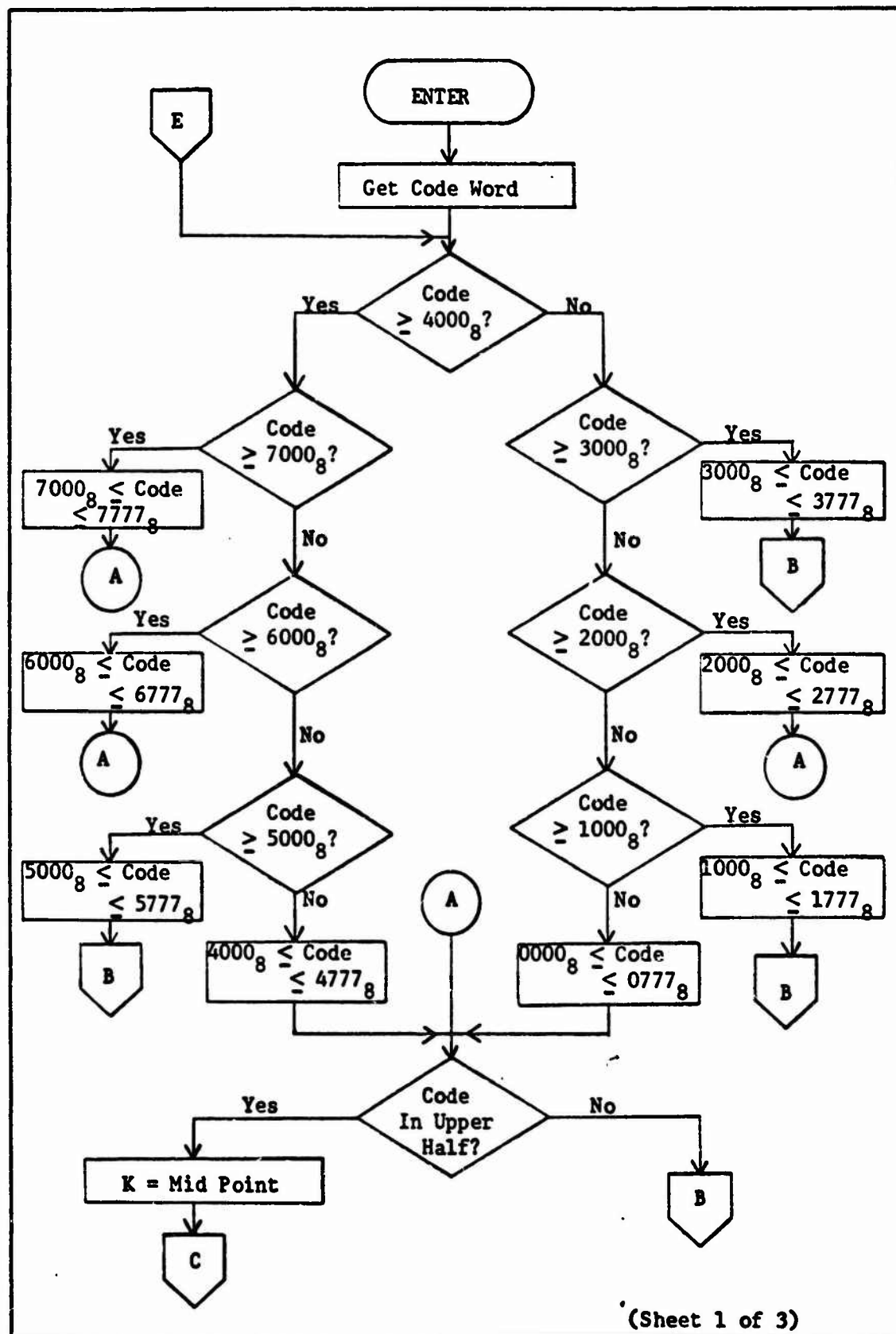


Fig. 4-10. Character Identification Routine Flowchart.

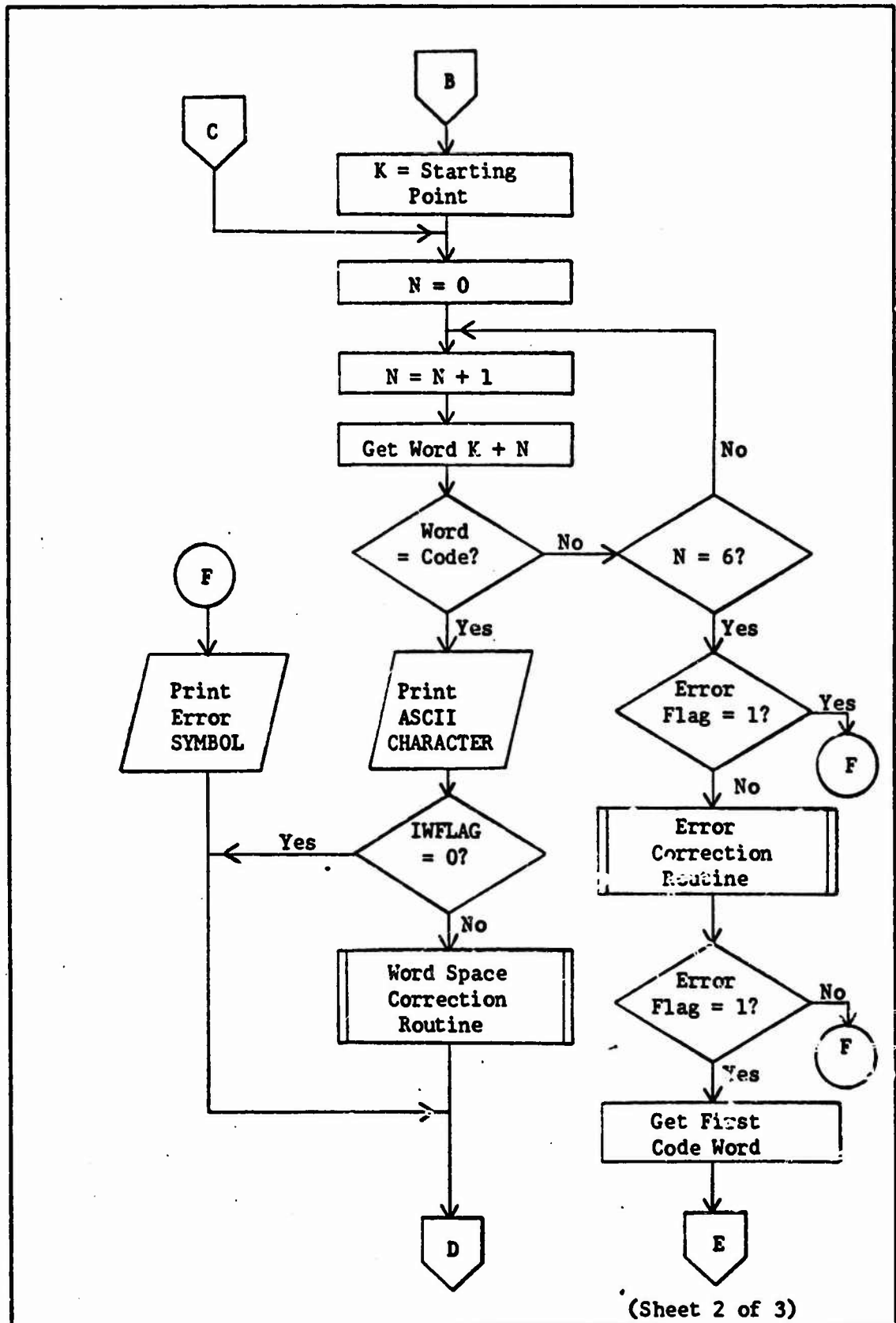
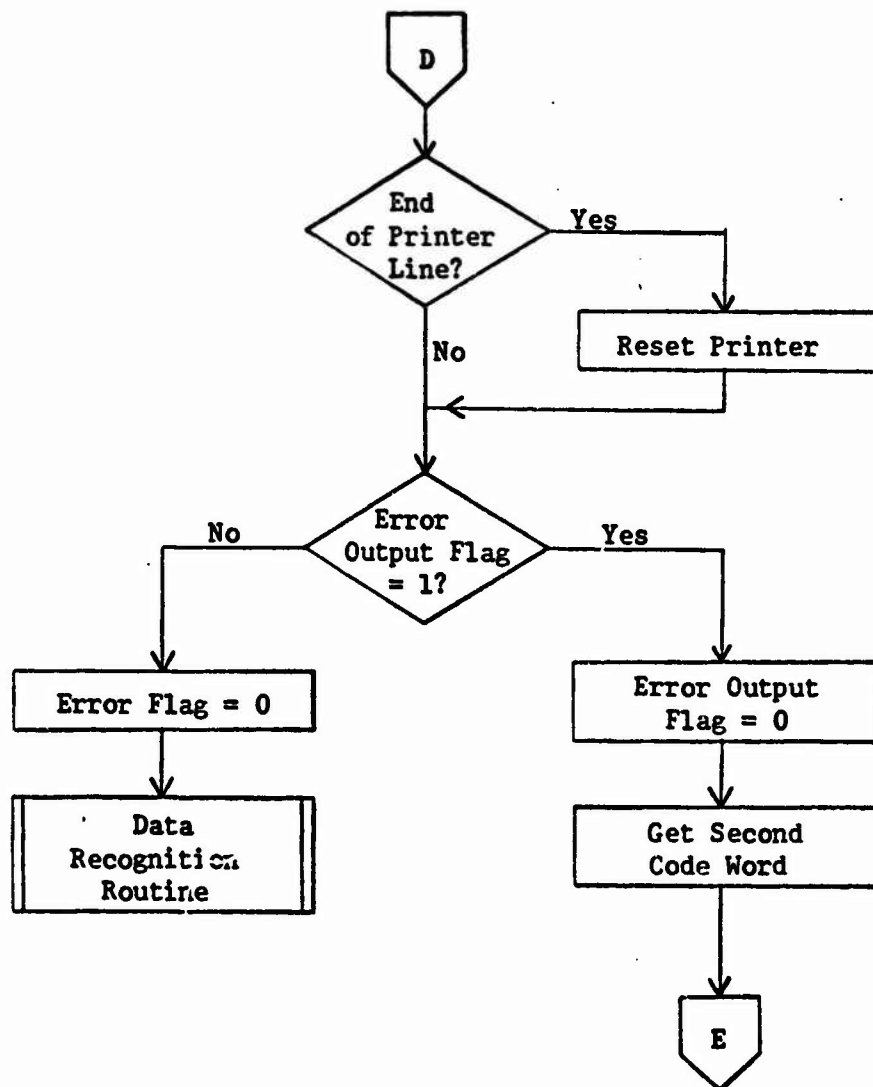


Fig. 4-10. Character Identification Routine Flowchart.



(Sheet 3 of 3)

Fig. 4-10. Character Identification Routine Flowchart.

Step 3 is only used for those subgroups having five or more code words. Finally, the fourth step identifies the correct internal code and corresponding ASCII code.

The identified ASCII character is then printed on the Teletypewriter. If a WORD space is identified, a blank is also printed. The printer carriage is controlled to provide double-spaced, 60-character lines.

If an internal code word cannot be identified, the Error Correction routine is entered. However, if the invalid code word is one which has been produced by the Error Correction routine, an error symbol is printed. Since it is possible for the Error Correction routine to generate two "corrected" code words, a network of flag checks is used to process each code word separately.

WORD Space Correction Routine. For the case of English language clear text transmissions, the occurrence of certain letters of the alphabet as the last character of a word is highly improbable, but not impossible. Six such letters are I, J, Q, U, V, and Z. When any of these characters is identified as the last character of a word, (i.e., followed by a WORD space), the WORD Space Correction Process is entered (Fig. 4-11). In this process, the word space is compared to a larger CHARACTER-WORD Space average than previously used. The adjusted average is the sum of the current CHARACTER-WORD Space average and the Pulse average, as shown below:

$$X = \frac{C-W \text{ SPACE AVG.} + \text{PULSE AVG.}}{2} - \frac{\text{WORD SPACE}}{2} \quad (4.15)$$

$$(X < 0 \Rightarrow \text{WORD SPACE}; X \geq 0 \Rightarrow \text{CHARACTER SPACE})$$

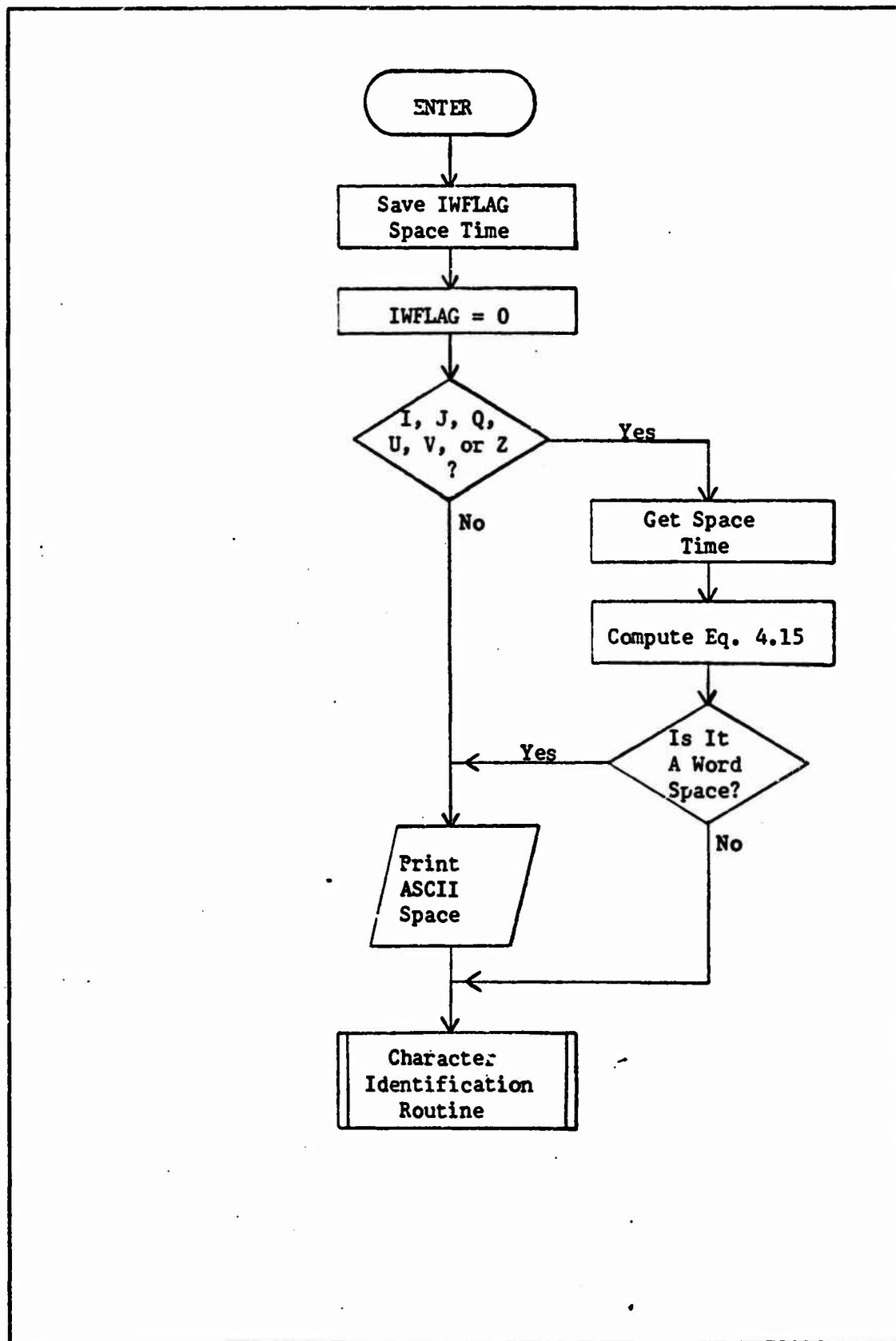


Fig. 4-11. Word Space Correction Routine Flowchart.

This calculation is used regardless of the type of preceding pulse.

If the space duration is greater than this adjusted average, it is still considered a WORD space. If, on the other hand, the space is less than the adjusted average, it is reclassified as a CHARACTER space instead of a WORD space.

The WORD Space Correction routine is applicable to English language clear-text transmissions only. If coded transmissions are received, this process may prove to be more of a hindrance than a help. The process is designed to correct mistakes on the part of the human sender, thus improving the readability of the recognition program output.

Error Correction Routine. The Error Correction routine (Fig. 4-12) is designed to correct errors due to either operator mistakes or signal noise. The Error Correction routine is entered only if the internal code word resulting from either of these two errors is unrecognizable by the Character Identification routine.

The Error Correction routine contains three successive parts, ordered by their relative importance to the process. In the first part, the number register is examined to determine if the maximum capability of 8 pulses has been exceeded in the invalid code word. If it has been exceeded, the Error Correction routine is exited and an error symbol is printed. If the limit has not been exceeded, the process advances to part 2.

The second part of the Error Correction routine is designed to eliminate extremely small DOT's caused by noise in the received Morse code signal. The pulses stored by the Data Recognition process are compared to a rejection value equal to one-half of the DOT average. The actual computation is:

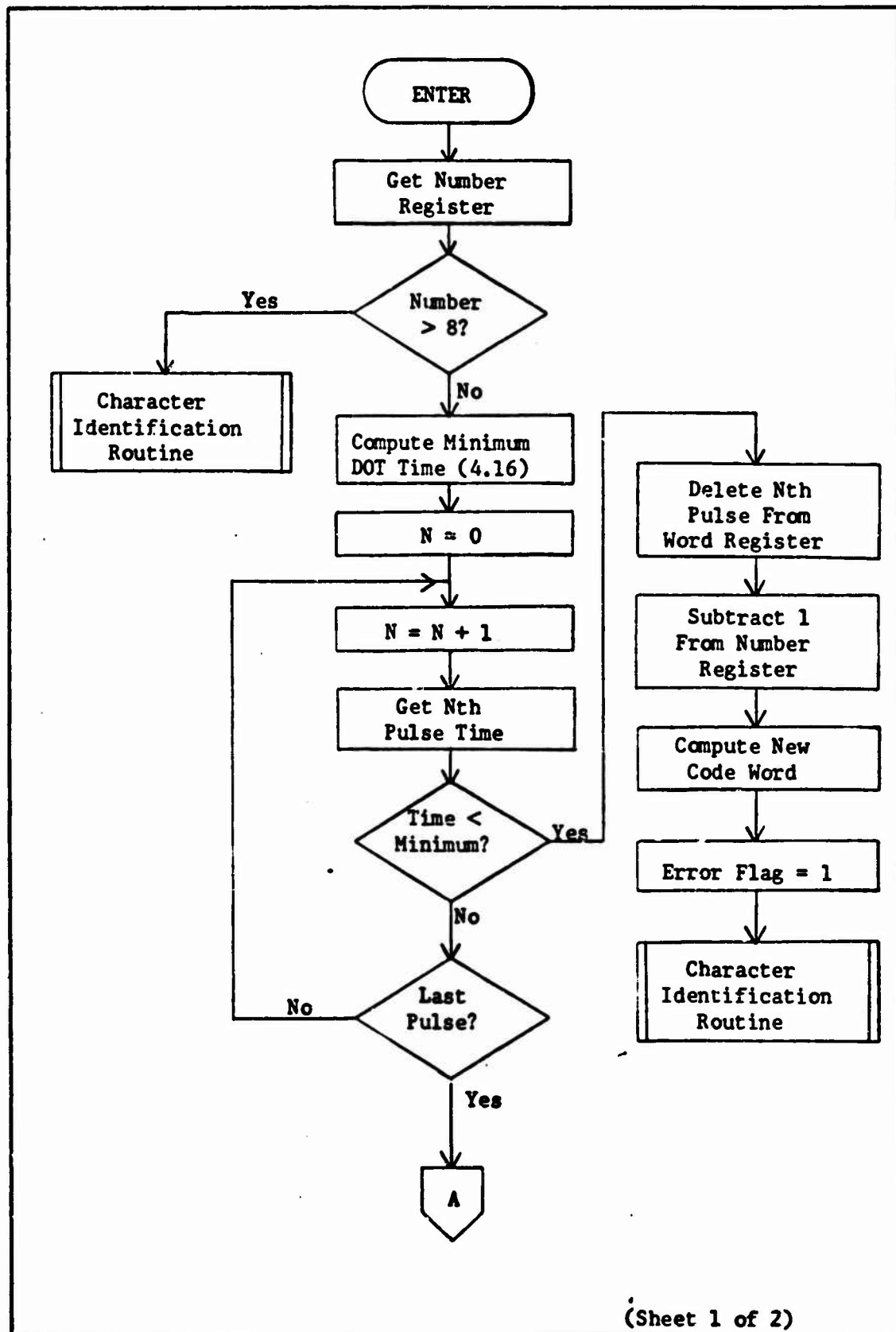
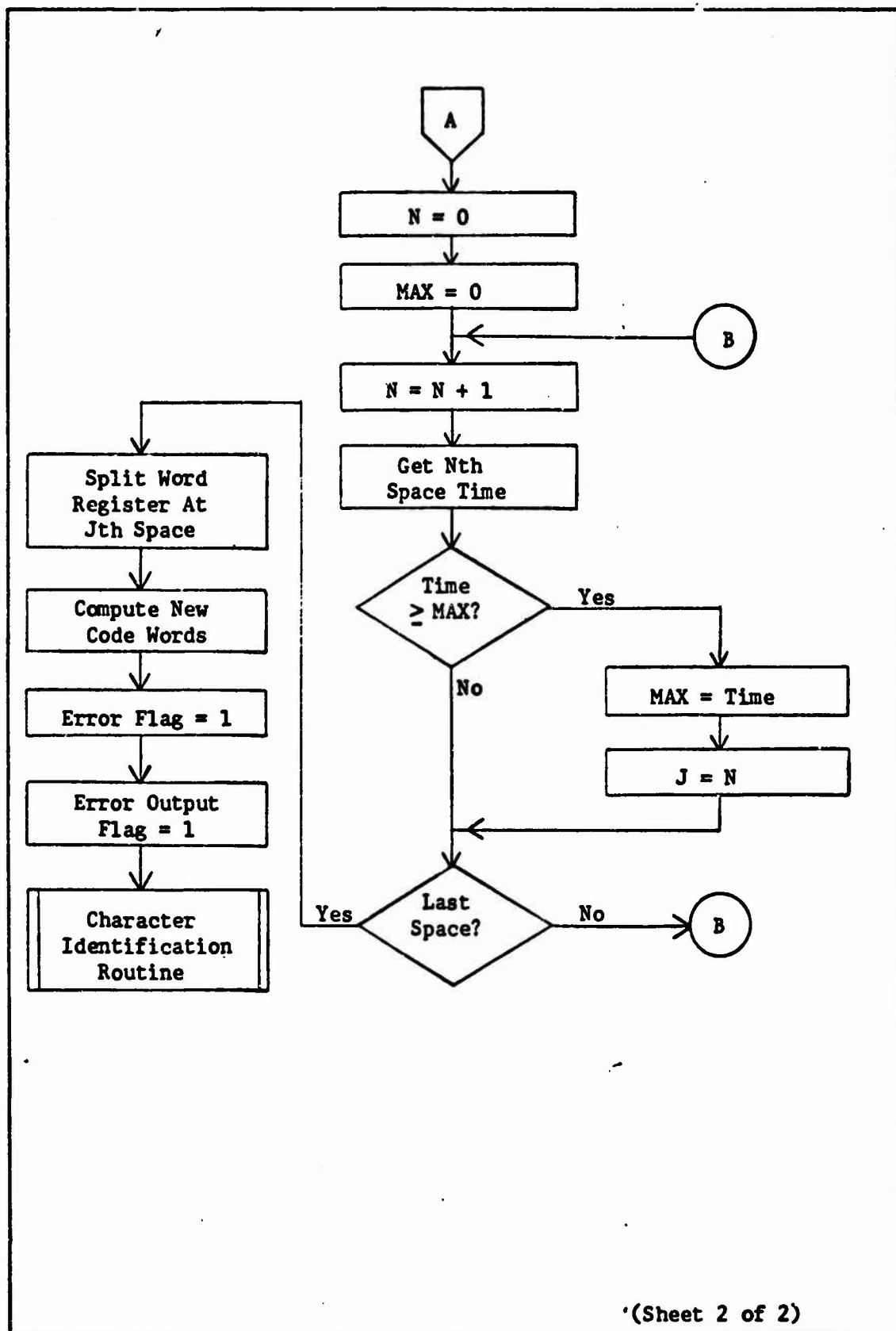


Fig. 4-12. Error Correction Routine Flowchart.



(Sheet 2 of 2)

Fig. 4-12. Error Correction Routine Flowchart.

$$\text{MINIMUM DOT TIME} = \frac{\text{DOT AVG.}}{2} \quad (4.16)$$

If none of the pulses have a time duration less than the minimum value, the process advances to part 3. If, however, one of the pulses is less than the minimum value, a new internal code is generated by eliminating the erroneous DOT. The corrected code is then processed by the Character Identification routine.

The third and final part of the Error Correction routine is designed to separate two run-on characters caused by a short CHARACTER space. This program operates on the assumptions that the invalid character is composed of two and only two run-on characters, and that the intended CHARACTER space is the longest of the SYMBOL spaces contained in the invalid character. Spaces stored by the Data Recognition process are examined to determine the largest space within the invalid character. Two new internal codes are then formed by separating the invalid character at that point. These codes are then processed one at a time by the Character Identification routine. In addition, if the space following the invalid character was identified as a WORD space, a blank is printed following the second new character, unless prohibited by the WORD Space Correction routine.

New code words generated by either part 2 or part 3 may not be identifiable by the Character Identification routine. Since the stored time durations for the original invalid character no longer apply to the new code words, entrance into the Error Correction routine caused by invalid new codes must be prohibited. This is done by setting appropriate flags for use in the Character Identification routine.

V. Operational Procedure

This chapter describes the operational procedure used to initialize, run, and reset the recognition program listed in Appendix A. This procedure applies specifically to the particular PDP-12 computer and associated peripheral devices used in this project, as shown in Fig. 5-1; however, the procedure will apply, in general, to other possible configurations utilizing the techniques presented in this report.

System Initialization

After the recognition program is loaded into the PDP-12 computer, several steps must be performed to initialize the system, as follows:

(Note: These steps can be accomplished in any order. Refer to Fig. 5-1 for equipment interconnections and control locations.)

1. Set the external Real-Time clock frequency. A value of 6000 Hz is best for hand-sent Morse code transmitted between 10 and 40 words per minute. Slightly higher or lower frequencies may be necessary for faster or slower transmission speeds, respectively.
2. Set the external program interrupt frequency. A value of 5000-6000 Hz is suggested; however, 6000 Hz is the maximum frequency that will allow proper operation of the Code Translation section. Higher program interrupt frequencies do not allow enough computer execution time for the Code Translation section to keep up with the output of the Signal Processing section.
3. Set the input signal threshold level. This value, set by rotating an A-D internal input channel control knob, has a range of $\pm 777_8$. A value of $\pm 777_8$ is suggested; however, any value above 400_8 will suffice. The threshold level is displayed on the CRT Display screen during the initialization process as a horizontal line. A setting of $+400_8$ corresponds to the line being positioned in the center of the screen; $+777_8$ corresponds to the top

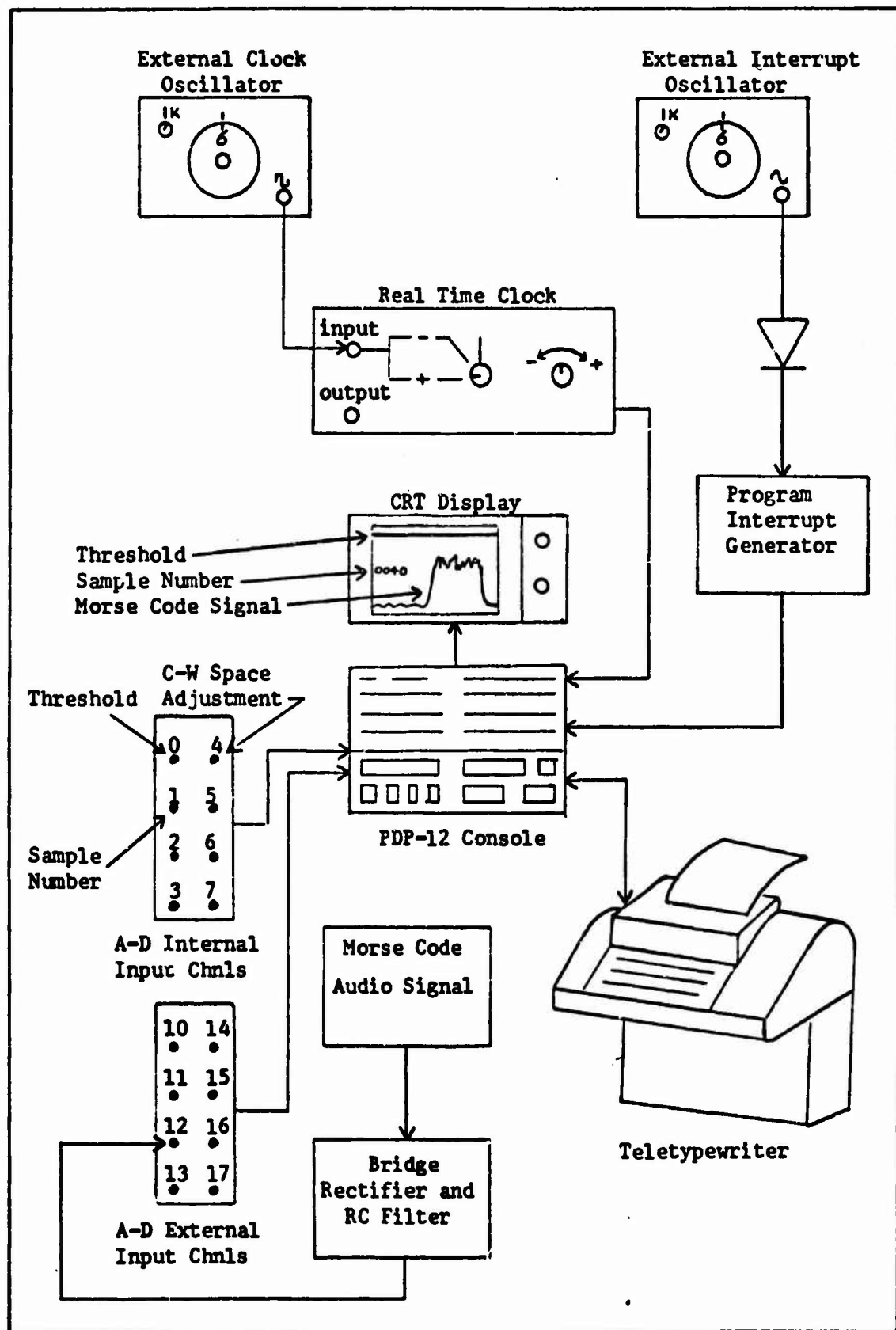


Fig. 5-1. Recognition Program Operational Configuration.

of the screen.

4. Set the peak rectified input signal voltage to approximately 2 volts. This signal is also displayed on the screen during the initialization process, provided a signal is being transmitted. The signal appears as a series of pulses flashing on the screen. As the input voltage level is increased, the peak of these pulses gets higher. The 2 volt level is reached when the pulse peaks disappear from view.
5. Set the number of input signal samples to be averaged. This number appears in the center left-hand side of the CRT Display screen during the initialization process. The number is set by rotating an A-D internal input channel control knob. An initial value of 0040_8 is recommended.
6. Set the CHARACTER-WORD space adjustment value to 0000_8 . This value is also set by an A-D internal input channel control knob. These control knobs have a 10-turn stop-to-stop movement. 0000_8 corresponds to the center of the knob movement.

When these six steps have been completed, the Recognition program is ready to process Morse code transmissions. Since the CRT Display only functions during the initialization process, the internal input channel control knobs should be calibrated for use during Recognition program operation. Recognition program operation is started by depressing any one of the Teletype keys.

Real Time Adjustments

The settings established in the initialization process will, in most instances, provide satisfactory recognition program output. Under unusual circumstances, however, certain readjustments may improve overall performance and readability of the printed program output. A list of possible programs and suggested readjustments is presented in Table III.

The presence of an extreme amount of interference on the Morse code

input signal will adversely affect recognition program performance. Such noise, generally having a time duration near that of a DOT or longer, causes the DOT, DASH, and pulse averages to be offset, resulting in an erroneous output. The respective averages will, however, be restored to normal when the interference subsides.

TABLE III

Recognition Program Real-Time Adjustments

<u>Problem</u>	<u>Solution</u>
1. Too many split words.	1. Increase the CHARACTER-WORD Space adjustment setting.
2. Too many run-on words.	2. Decrease the CHARACTER-WORD Space adjustment setting.
3. Too many error symbol printouts (a) Clear Morse Code Signal (b) Noisy Morse Code Signal	3. (a) Increase or decrease the Sample Number setting to obtain the best output. (b) Lower the peak input signal voltage slightly and readjust Sample Number setting.
4. Long succession of E's or I's.	4. Most likely due to extremely noisy Morse Code signal. Increase Sample Number setting and perform restart procedure.

Restart Procedure

The restart feature provides the capability to return to the System Initialization point at the beginning of the recognition program. This is done by depressing and resetting (rocking) Sense Switch 1 on the PDP-12 console at any time during program operation. Program operation may be resumed again by depressing any one of the Teletype keys.

The restart feature provides two important capabilities to the program. First, it erases current average information. This is most

useful in the case where the averages are offset due to extreme signal interference. By performing a quick restart procedure (i.e., rocking Sense Switch 1 followed immediately by depressing a Teletype key), the erroneous averages are erased and new averages are established. Morse code information is lost only during the brief interval between the two actions.

Second, the restart procedure permits the last character of a Morse code transmission to be printed out. Normally, a Morse code character is not identified until either a CHARACTER space or WORD space is recognized. These spaces are recognized, as are all pulses and spaces, by their relative time durations. Space time durations are obtained when the succeeding pulse begins. If the pulse never occurs, as is the case at the end of a transmission, the last space time is never obtained and, as a result, the last Morse code character is not processed. Depression of the restart switch (Sense Switch 1) causes all unprocessed words stored in the 200₈-word memory buffer to be processed, including the last Morse code character received.

VI. Results

This chapter presents an evaluation of recognition program performance. Various types of Morse code transmissions were used to test the program. These include hand-sent Morse code transmitted via a simple hand key, a bug, and an electronic keyer, as well as machine-sent Morse code (Kleinschmidt-paper tape). The effects of noise on the performance of the recognition program was also tested.

The types of errors made by the program were mainly due to characteristics peculiar to the sender. A classification of the errors encountered during program testing is given along with error rates and program limitations.

Man vs. Machine Comparison

The method by which the performance of the recognition program should be evaluated is open to question. To base the evaluation strictly on a comparison of human recognition versus machine recognition of a Morse code transmission is unfair. The following excerpts from Gold's paper (Ref 4:18) makes this point very clear:

Morse code is itself not a language but a way of representing or coding a given language, such as English or German. It is analogous to handwriting in that there is a symbol for each character in an alphabet. Some people write very clearly so that anyone can read their writing and, further, even those who cannot understand the language of the clear writer may still be able to understand and reproduce each symbol (letter of the alphabet) that was written. It is, however, very helpful to know the language being written; in fact, it is usually more difficult to read handwriting in a foreign language, even if this language is somewhat familiar, than to read one's native language in the same handwriting. Thus we see that, for handwriting, knowledge of the code

is sometimes (in the case of the clear writer) sufficient to discern each intended symbol; in many cases, a high knowledge, i.e., of the language, is necessary to do so. Note that understanding of the meaning is not being discussed, but merely the identification, symbol for symbol, of what was written. ...it was noted that a man, who knows Morse code, will never remember a sample of data as accurately as a machine; yet he decodes a Morse message well enough to make any machine thus far built turn green with envy. Why? The answer can only be that to a very great extent he knows what to expect because he knows Morse code and the language being sent.

Thus we see that man's knowledge of the language being sent give him a tremendous advantage over a machine, unless the machine has a similar knowledge of the language. The machine described in this project does not have this knowledge and should not be evaluated on a strict man versus machine basis.

However, it is permissible to evaluate performance on a man versus machine-plus-man basis. That is, compare man's ability to recognize a Morse message with man's ability to interpret the message recognized by the machine. Again, man has the decided edge in this comparison. However, man must be able to interpret the output of the machine or the machine is of no value at all. The following excerpt from Gold's paper (Ref 4:22) gives his point of view on this subject:

It is felt that the effectiveness of a machine... depends on the percentage of received messages that it can adequately decode. The word "adequately" pinpoints the vagueness of this statement; this is interpreted to mean "capable of reconstruction in a reasonable length of time;" even here the problem becomes too subjective.

Testing Procedure

The performance of the recognition program was evaluated in the

following manner. Tape recordings of hand-sent Morse code transmissions were used as the input signal to the machine. These tapes contain examples of messages transmitted by hand key, semi-automatic key ("bug"), electronic keyer, and machine. The type of messages recorded include plain text and code groups. Transmission speeds vary from 7 words per minute (wpm) to 35 wpm. In all, seven different recording sessions were used to evaluate machine performance. Table IV lists the key features and error rates for each recording session.

A 550-word plain text was used as the transmitted message for Recording Sessions 1 through 4. The prepared text and the recognition program printout for each of these recording sessions is contained in Appendix D, Figures D-1 through D-5. On all but Recording Session 4, a copy of the prepared text was given to two people listening to the Morse code transmission. These people annotated their copy to indicate transmission mistakes made by the sender. The two annotated copies were then used to identify recognition program errors (i.e., discrepancies between the prepared text and the machine output) due to the sender and not the machine itself. The remaining errors were used to approximate machine error rate percentages.

All recognition program errors, whether due to sender error or the machine itself, have been classified and are presented in the following section.

Classification of Errors

A classification of the errors encountered in recognition program translations of hand-sent Morse code plain text transmissions is now presented.

TABLE IV
Recording Session Variables and Error Rates

<u>Recording Session</u>	<u>Sending Unit</u>	<u>Transmission Rate (wpm)</u>	<u>Type of Message</u>	<u>Character Errors/ Characters Sent</u>	<u>Error Percentage</u>	<u>Remarks</u>
1	"Bug"	16	Plain Text	8 ^a /2927	0.273	Sending Unit Malfunction
2	Hand Key	12	Plain Text	5 ^a /2927	0.171	
3	"Bug"	18	Plain Text	43 ^a /2927	1.47	Sending Unit Malfunction
4	Electronic Keyer	35	Plain Text	45 ^b /2927	1.53	
5	Electronic Keyer	7	Plain Text/ Code Groups	0 ^b /825	0.00	
6	Machine	18	Plain Text	44 ^b /633	6.9	Radio Broadcast Interference and Signal Fading
7	Hand Key	8-10-12	Code Groups	3 ^b /914	0.33	"white" noise

a Machine errors only

b Total errors

- 1) Two Characters run together. Examples: ME → G, TH → 6,
NA → X. Caused by too short a space separating the characters.
- 2) Split characters. Examples: Y → TW, V → ST. Caused by too
long a space separating pulses within a character.
- 3) Extra DOT (valid code). Examples: W → AA, Y → TU, C → ND.
Caused by transmission of an extra DOT. The resulting character,
in this case, is not a valid one. The two printed characters
are the product of the Error Correction routine.
- 5) Missing DOT (valid code). Examples: H → S, C → K, R → A.
Caused by absence of a required DOT.
- 6) Missing DOT (invalid code). Examples: Period → AK or CT,
9 → MM. Caused by absence of a required DOT resulting in an
invalid code. The two printed characters are a result of the
Error Correction routine.
- 7) Simple pulse errors. Examples: Y → TU, K → O, C → B. Caused
by an extra-long DOT being recognized as a DASH, or an extra-
short DASH being recognized as a DOT. Occurs most frequently
when a hand key is used. One or two characters may be printed,
depending on whether the resulting internal code word is valid
or not.
- 8) Complex pulse errors. Examples: B → K, O → X, B → M. Caused
by two or more DOTs, spaced closer together than usual, being
recognized as a DASH. Also caused by a DASH being broken up into
two or more DOTs. Occurs most frequently in the presence of
input signal noise.
- 9) Space errors. Examples: all run-on or split words. Caused by
either too short a WORD space or too long a CHARACTER space,

respectively. Consistant run-on or split words are caused by an incorrect CHARACTER-WORD Space Adjustment control knob setting.

10) Complex errors. Indicated by the error symbol (=) printout.

The source of these errors cannot be traced. The invalid code, resulting from an unknown source, could not be corrected by the Error Correction routine.

Errors of the type 3) through 6) occurred most frequently when a semiautomatic key was used. This can be explained by noting that the number of DOTs transmitted in succession is a function of how long the DOT lever is depressed on these types of sending units. Occasionally the sender, through fatigue or carelessness, sends too many or too few DOTs by not depressing the lever for the correct amount of time.

Error type 8) occurred quite frequently during Recording Sessions 1 and 3. The cause of these errors was traced to improper adjustment of the DOT-sending portion of the "bug" used for both transmissions. Indeed, both recording sessions had to be stopped several times to allow readjustments to be made on the sending unit. Noise induced by contact bounce as a result of the malfunctioning unit was the primary cause of the type 8) errors experienced.

Recording Session 3 contained many type 2) errors. In almost every instance where a V (. . . -) was transmitted, the recognition program printed an S (. . .) T (-) instead. The SYMBOL space following the third DOT was misinterpreted as a CHARACTER space by the machine. Fig. 3-4 (Chapter III) reveals the cause of these machine errors. Notice that the DOT-SYMBOL cluster extends above the SYMBOL linear decision boundary, and that the SYMBOL space variance is much larger in this cluster than it is in corresponding clusters for Recording Sessions

1 and 2 (Figures 3-2 and 3-3 respectively).

There are two possible explanations for this abnormally large SYMBOL space variance: 1) "bug" contact bounce causing the time duration of the last automatically generated DOT to be shorter than usual and, correspondingly, the time duration of the following space to be longer than usual, and 2) too fast a transmission rate setting on the automatic DOT generator, as compared to the actual transmission rate of the message itself. The latter of these two possible explanations is the most feasible, since the DOT portion of the "bug" had a very restricted transmission rate adjustment range over which it would satisfactorily operate. In fact, readjustment was required several times during the 30-minute transmission period. It is quite possible, then, that the actual transmission rate used was too fast, i.e., faster than the rate of the manually transmitted portion of the message. Under these circumstances, the time duration of automatically generated SYMBOL spaces would be shorter than the time duration of manually generated SYMBOL spaces, thereby accounting for the wide time duration variance, and the resulting abundance of type 2) errors.

Analysis of Performance

Recognition program performance was analyzed by comparing the printouts obtained for Recording Sessions 1, 2, and 3 with the annotated worksheets provided by the human receivers. Errors identified on the worksheets were marked on the corresponding machine printout to eliminate them from consideration in the error analysis. These errors are marked with an asterisk on the machine printouts shown in Appendix D. The remaining discrepancies contained in the machine printouts were noted

and used to compute the error rate percentages given below. These errors are marked with a number symbol (#) on the printouts.

Error Rates. The error rates calculated for Recording Sessions 1, 2, and 3 by the previously described method are given in Table IV. Note that error rates are given for character errors only. Space errors causing run-on or split words, although having an effect on the readability of the message, are not considered essential to the overall performance of the recognition program. Character error rate percentages were calculated by dividing the number of incorrect characters noted on the machine printout by the total number of characters in the transmitted message.

The error rate indicated for Recording Session 4 was calculated by dividing all the "uncorrected" character errors by the total number of characters transmitted. "Uncorrected" errors are those machine printout errors occurring in words which were not retransmitted by the operator. During this transmission, the operator followed the standard practice of retransmitting words in which a mistake was made and noted by the operator. The error rates for Recording Sessions 5, 6, and 7 were calculated by dividing all machine printout errors by the number of characters transmitted.

It is interesting to note that many errors annotated by the human receivers did not occur on the machine printout. This, of course, is due mainly to the Error Correction feature of the recognition program. The Error Correction routine was used a total of 15 times during Recording Session 1, 5 times during Recording Session 2, and 5 times during Recording Session 3.

Limitations. Recording Sessions 4 through 7 provide a means of

determining the operational limitations of the recognition program. Knowledge of these limitations is necessary to determine the weak points of the program and to try to correct them.

In general, the recognition program performs well for all types of Morse code transmissions. The major weakness is input signal noise; especially interference having a time duration large enough to alter the DOT, DASH, and pulse averages used by the program to establish linear decision boundaries in two-dimensional pattern space. Although the respective pulse averages do return to their correct values when the interference subsides, the program output during the interference period is, in most cases, unintelligible.

A brief discussion on the results obtained for Recording Sessions 4 through 7 will now be given. Recording Sessions 4 and 5 were transmitted by the same person using the same equipment. Recording Session 4 (Fig. D-5) was transmitted at varying rates ranging from the individual's maximum rate of 35 wpm to a low of approximately 15 wpm. Most of the text was transmitted at the 35 wpm rate. The operator, by his own admission, was sending at his maximum capability and made many mistakes, not all of which were noticed and corrected by the repetition procedure previously described.

Recording Session 5 (Fig. D-6) consists of sentences and code groups used in teaching Morse code to a group of students. With the exception of one sentence, all transmissions were sent one character at a time at a steady 7 wpm rate. This explains the spaces appearing between each character on the machine printout. No machine errors occurred during this recording session.

Recording Session 6 (Fig. D-8) is a radio broadcast of an American

Radio Relay League W1AW cw bulletin, transmitted by machine at 18 wpm. A copy of the transmitted bulletin is shown in Appendix D, Fig. D-7. Much interference and signal fading occurred during this recording session, although not severe enough to prevent a human from correctly copying the message. These disturbances, however, had a great affect on the performance of the machine, as is indicated on the machine printout.

Recording Session 7 (Fig. D-9) consists of code groups transmitted at 8, 10, and 12 wpm on a hand key. These transmissions were obtained from tape recordings of hand-sent Morse code used by the U.S. Army to teach Morse code during WWII. The tapes contained a background "white" noise which did not, as the machine output shows, affect recognition program performance.

Analysis of all results obtained from the seven recording sessions indicates that the recognition program is capable of recognizing hand-sent Morse code at speeds up to at least 35 wpm with moderate signal noise. The main limitation thus far discovered in this project is pulse-type interference.

VII. Conclusions and Recommendations

This chapter presents the conclusions reached from the limited testing performed on the recognition program. Also, recommendations concerning further testing procedures and possible program changes are given.

Conclusions

Although only seven different types of Morse code transmissions were used to evaluate the performance of the recognition program, the results obtained clearly define the attributes and weaknesses of the program in its present form. Certainly more work must be done on the Signal Processing section to limit the detrimental effects of input signal pulse-type interference, since signal interference is prevalent on the radio bands used for cw transmissions. Perhaps a simpler algorithm than that of the low-pass digital filter technique used in the program will produce better results. A combination of digital filtering and other noise reduction techniques might prove to be optimum.

Aside from the interference problem, the performance of the recognition program is very satisfactory. The character error rates calculated for Recording Sessions 1, 2, and 3 are well within an acceptable limit. The readability of the machine output, a function of the particular sender, is optimizable by real-time adjustment. Since it is desirable to have the machine operate with as little human intervention as possible, preferably none, procedures are given in the next section to automate the readability adjustment as well as certain other controls now requiring real-time adjustments to maximize performance.

These procedures and recommendations are now presented.

Recommendations

The recommendations presented in this section deal with recognition program improvements and hardware realization suggestions.

Recognition Program Improvements. The recognition program, in its present form, represents an attempt to perform machine recognition of hand-sent Morse code by methods other than those previously documented. Admittedly, when the project was undertaken there was no guarantee that the techniques used in this program would be as good as or better than those other methods.

The use of averaging algorithms can be very misleading. Extreme care must be exercised when trying to fit an averaging technique to a particular physical phenomenon. For example, the average human being has one testicle and half a uterus. Basing decisions on this type of average is certainly absurd. However, the distribution properties of hand-sent Morse code indicate that averaging techniques are indeed applicable. Test results confirm this conclusion.

Unfortunately, the use of averaging techniques presents a problem when pulse-type interference is present in the input signal. Interference of this type offsets the pulse averages used in the program. One possible solution to this problem is the establishment of upper and lower bounds around the DOT and DASH averages computed in the program. Data distribution plots indicate a relatively tight grouping of DOT and DASH time durations for a particular sender. These time durations may be assumed to have a normal distribution with a mean approximated by the calculated averages (Ref 3:27). A 2-sigma confidence interval can easily

be established and used to exclude time durations not within this interval from the averaging process. This method should be capable of detecting time duration average differences present in a two-way conversation. The data distribution plots reveal that time duration intervals remain within a 2-sigma interval of each other for transmission rates between 10 and 20 wpm. Quite possibly this overlap exists for an even larger difference in transmission rates. It is recommended that additional data be obtained to confirm or deny this possibility.

The human intervention now required to optimize readability of the machine printout can easily be eliminated by an additional program routine. The number of spaces printed on each line of machine output can be compared with an upper and lower limit to determine whether adjustment is needed. For example, if the number of spaces printed exceeds 15, the CHARACTER-WORD space adjustment should be increased; if the number of spaces is less than 5, the adjustment should be decreased. The amount of readjustment per change can be optimized to prevent over correction while still allowing the optimum value to be reached within a reasonable amount of time. The upper and lower limits and the incremental adjustment value must be determined experimentally.

Human intervention now necessary to optimize the number of input signal samples to be averaged for the change detection process can also be replaced by computer programming. An initial value, preset at the start of the program, can be adjusted to an optimum value as a function of the DOT average. For example, the value can be adjusted to filter out all pulses having a time duration of less than one half the DOT average. The low-pass cut-off frequency of the digital filter is determined by the number of samples averaged and the interrupt frequency.

Since the interrupt frequency is a constant, set at the maximum value permitting proper Code Translation section operation, the relationship between number of samples and cut-off frequency is linear. The number of samples to be averaged is thus directly proportional to the DOT average. The actual relationship must be determined experimentally.

Hardware Realization. Many of the peripheral devices and control knobs used in this project can be eliminated in a hardware realization of the recognition program. The input signal threshold level can be set at a constant value of $+777_g$, thereby eliminating the threshold control knob. A feedback control circuit can be used to regulate the peak dc voltage of the rectified input signal to an optimum value for the constant threshold. Inclusion of the program changes recommended in the previous section would eliminate the need for sample number and readability control knobs. It may be desirable, however, to retain these controls to permit human intervention if required under unusual circumstances. The two switches used to perform a program restart can be replaced by one START/RESET pushbutton.

The entire program, now requiring about 3K of memory, can easily be reduced to fit within a 1K 12-bit memory unit by elimination of those instructions now used to control the CRT Display and sample switches. Elimination of these instructions has the added benefits of faster program execution time, better reliability, and less construction cost.

The use of a 16-bit structure will provide additional programming capabilities not possible with the 12-bit structure of the PDP-12 computer. First of all, a constant Real Time Clock frequency can be used for all Morse code transmission rates, since the 16-bit word length will provide a 65,536 (2^{16}) clock counter overflow range. For example, an external

clock frequency of 65,536 Hz will permit recognition of pulse and space time durations ranging from 1 or 2 milliseconds to one second in length. This corresponds to transmission rates in the range of 5 to 500 wpm.

The 16-bit structure will also permit increased error correction capability. The 12-bit structure used in the recognition program permits error correction of a run-on Morse characters not exceeding 8 pulses in length. The 16-bit structure will permit correction of run-on characters up to 12 pulses in length. Effective use of this increased capability will require revision of the Error Correction routine to allow for the possibility of three run-on characters.

BIBLIOGRAPHY

1. American Radio Relay League. The Radio Amateur's Handbook (Forty-Seventh Edition). Newington, Conn.: American Radio Relay League, Inc., 1970.
2. Digital Equipment Corporation. PDP-12 System Reference Manual (Fourth Printing). Maynard, Mass.: Digital Equipment Corporation, October 1971.
3. Freimer, M., B. Gold, and A.L. Tritter. "The Morse Distribution." Institute of Radio Engineers Transactions on Information Theory, 1:25-31 (March 1959).
4. Gold, B. "Machine Recognition of Hand-Sent Morse Code." Institute of Radio Engineers Transactions on Information Theory, 1:17-24 (March 1959).
5. Petit, R.C. "The Morse-A-Verter - Copying Morse Code by Machine." QST Vol. LV Number 1:30-35 (January 1971).
6. Powers, B.L., and F.R. Scalf. The Design of a Morse-to-Teletype Signal Converter Using Integrated Micrologic Circuitry. Unpublished thesis. United States Naval Postgraduate School, June 1968.
7. Ullman, J.R. Pattern Recognition Techniques New York: Crane, Russak and Co., Inc., 1973.
8. West, L.J. Review of Research on Morse Code Learning - Summary of Findings and Recommendations. Technical memorandum. Chanute Air Force Base, Illinois: Training Aids Research Laboratory, November 1968.

Appendix A

Recognition Program Listing

The Recognition Program was written for use on the PDP-12 computer equipped with a CRT Display, Teletype or DECWRITER, KW12-A Real Time Interface, Data Terminal, and a TU56 Dual Drive Transport. The program is written in LAP6-DIAL assembly language.

The listing contains six columns of information. These are, from left to right, program line number, memory field location, octal value assigned to the location, user-assigned symbols, program and assembly instructions or data, and comments.

0000			*20		
0001			/	HAND SENT MORSE CODE	
0002			/	RECOGNITION PROGRAM	
0003			/		
0004			/		
0005			/		
0006			/		
0007				PMODE	
0010				*0	
0011	0000	0000	0		/PMODE
0012	0001	5402	JMP I 2		/INTERUPT
0013	0002	0233	SERVE		
0014			*5		
0015	0005	0000	0		
0016	0006	0000	0		
0017	0007	0000	0		
0020	0010	0000	0		
0021	0011	6000	SIGNAL.	6000	
0022	0012	0577	STORE.	577	
0023	0013	0577	WORD.	577	
0024	0014	4017	STORIT.	4017	/ASCII MEMORY
0025			*17		
0026	0017	7767	7767		
0027			*20		
0030	0020	0000	CHECK.	0	
0031	0021	0000	NUM.	0	
0032	0022	0000	COUNT.	0	
0033	0023	0400	SADISO.	SADIS	
0034	0024	1200	ADSAMO.	ADSAM	
0035	0025	1600	PROCEO.	PROCES	
0036	0026	0011	KCLP.	0011	
0037	0027	0000	CHANGE.	0	
0040	0030	0271	SENSE0.	SENSE	
0041	0031	7001	M777.	-777	
0042	0032	0577	K577.	577	
0043	0033	0000	NUMR.	0	
0044	0034	2214	EFL10.	EFL1	
0045	0035	2251	EFL20.	EFL2	
0046	0036	0000	EFLAG1.	0	
0047	0037	0000	EFLAG2.	0	
0050			LMODE		
0051	0040	0000	0		/LMODE
0052	0041	6221	JMP SERVEL		/INTERUPT
0053			PMODE		
0054	0042	2300	SETPT0.	SETPTR	
0055	0043	0000	MSTART.	0	
0056	0044	0000	WDREG.	0	
0057	0045	2400	IPROC0.	IPROC	
0060	0046	0000	CODE.	0	
0061	0047	3000	C01.	AUNK	
0062	0050	3014	C02.	A5	
0063	0051	3026	C11.	ASPAC	
0064	0052	3036	C21.	AU	

0065	0053	3044	C22.	AWAIT
0066	0054	3052	C31.	A
0067	0055	3062	C41.	W
0070	0056	3072	C42.	A6
0071	0057	3102	C51.	T
0072	0060	3114	C61.	K
0073	0061	3122	C62.	A7
0074	0062	3130	C71.	M
0075	0063	3134	C72.	A9

0076	0064	3202	CHKIT.	CHCK
0077	0065	3013	CN5.	N5
0100	0066	3043	CWAIT.	WAIT
0101	0067	3071	CN6.	N6
0102	0070	3121	CN7.	N7
0103	0071	3133	CN9.	N9
0104	0072	2607	ASCO.	ASCII
0105	0073	1602	PRCAG0.	PRCAGN
0106	0074	7700	KLINE.	7700
0107	0075	7700	LINE.	7700
0110	0076	0000	IWFLAG.	0
0111	0077	7777	M1.	-1
0112	0100	0001	K1.	1
0113	0101	1777	K1777.	1777
0114	0102	0000	PX.	0
0115	0103	7140	M640.	-640
0116	0104	0000	DOTAV.	0
0117	0105	0000	DASHAV.	0
0120	0106	0000	PAVG.	0
0121	0107	0000	FUDGE.	0
0122	0110	0000	SAFUDG.	0
0123	0111	0000	HOLDWD.	0
0124	0112	0377	K377.	377
0125	0113	0000	TEMP.	0
0126	0114	0240	K240.	240
0127	0115	0215	K215.	215
0130	0116	0212	K212.	212
0131	0117	0100	RATE0.	0100
0132	0120	0000	CWAVG.	0
0133	0121	0013	K13.	13
0134	0122	7776	M2.	-2
0135	0123	7776	KM2.	-2
0136	0124	2050	PWD0.	PWD
0137	0125	2200	ICODE0.	ICODE
0140	0126	0217	RSTAP0.	RSTART
0141	0127	6000	K6000.	6000
0142	0130	0000	PFLAG.	0
0143	0131	0000	KCHANG.	0
0144	0132	0430	COND10.	COND15
0145	0133	2000	SDASH0.	SDASH
0146	0134	2254	DIAL0.	DIAL
0147	0135	0777	K777.	777

0150	0136	7777	PAST,	7777	/INTL VAL = -1
0151	0137	0000	THRESH,	0	
0152	0140	1443	ADOUT0,	ADOUT	
0153	0141	1404	RESET0,	RESET	
0154	0142	1400	TIMER0,	TIMER	
0155	0143	1332	ADSAM1,	ADSAM2	
0156	0144	0000	NEW1,	0	
0157	0145	0000	PORS,	0	
0160	0146	0000	WHICH,	0	
0161	0147	0000	TIME,	0	
0162	0150	0000	INF1,	0	
0163	0151	0000	INF2,	0	
0164	0152	0000	NUM31,	0	
0165	0153	0000	NUM32,	0	
0166	0154	4017	KSTOR,	4017	
0167	0155	0000	UNFLG,	0	
0170	0156	0000	UNCK1,	0	
0171	0157	0000	UNCK2,	0	
0172	0160	0000	UNCK3,	0	
0173	0161	0000	ERWDNM,	0	
0174	0162	3400	UNKCH0,	UNKCHK	

0175	0163	0000	TEMPMN,	0	
0176	0164	0000	MNUM,	0	
0177	0165	0000	LMNUM,	0	
0200	0166	1500	FIG10,	FIG1	
0201	0167	3467	RFIG10,	RFIG1	
0202	0170	1516	FIG20,	FIG2	
0203	0171	3600	RFIG20,	RFIG2	
0204	0172	0000	TALLY,	0	
0205	0173	0000	TALLY1,	0	
0206	0174	0000	TALLY2,	0	
0207	0175	2404	IPRTN0,	IPRTN	
0210	0176	1723	CALSP0,	CALSPA	
0211	0177	0000	CDE32,	0	
0212				SHL=7413	
0213				CAM=7621	
0214					
0215					
0216				LMODE	
0217				KCC=6032	
0220				KSF=6031	
0221				RMF=6244	
0222				ION=6901	
0223					
0224					
0225				PMODE	
0226					
0227					
0230					
0231					
0232					

0233				*200	
0234	0200	7300	START,	CLA CLL	
0235	0201	1127		TAD K6000	
0236	0202	6132		CLLR	/EXTNL CLK RATE
0237	0203	7200		CLA	
0240	0204	1322		TAD CFLAG	
0241	0205	6134		CLEN	/ENABLE INPUT 1
0242	0206	7200		CLA	
0243	0207	6046		TLS	
0244	0210	6032		KCC	
0245	0211	6141		LINC	
0246				LMODE	
0247	0212	1020		LDA I	
0250	0213	0240		0240	
0251	0214	0004		ESF	/FULL SZ DSPLY &
0252	0215	0002		PDP	/DSABL TTY INTRP
0253				PMODE	
0254	0216	4442		JMS I SETPT0	/CR + 2 LFS
0255	0217	4423	RSTART,	JMS I SADIS0	/SET AD SMPL SIZ
0256	0220	5425		JMP I PROCE0	/GO TO MAIN PROG
0257			/		
0260			/		
0261			/		
0262			/		
0263			/	INTERUPT SERVICE ROUTINE	
0264			/		
0265			/		
0266				LMODE	
0267	0221	4325	SERVEL,	STC AC	/SAVE AC
0270	0222	0261		ROL I 1	
0271	0223	4323		STC LINK	/SAVE LINK
0272	0224	2040		ADD 0040	/GET RTN ADDRSS
0273	0225	1620		BSE I	
0274	0226	6000		6000	
0275	0227	4270		STC LRTN	/CREATE RTN JMP
0276	0230	2100		ADD K1	
0277	0231	4326		STC LFLAG	/SET LFLAG
0300	0232	6241		JMP LCONT	
0301				PMODE	
0302	0233	3325	SERVE,	DCA AC	/STORE AC VALUE
0303	0234	7204		GLK	
0304	0235	3323		DCA LINK	/STORE LINK
0305	0236	1000		TAD 0000	
0306	0237	3324		DCA HOLDIN	/STORE PC
0307	0240	6141		LINC	
0310				LMODE	
0311	0241	0643	LCONT,	LDF 3	/SET DF = 3
0312	0242	0002		PDP	
0313				PMODE	
0314	0243	4424		JMS I ADSAM0	/YES GO TO ADSAM
0315	0244	7300	EXIT,	CLA CLL	

0316	0245	1326		TAD LFLAG	
0317	0246	7640		SZA CLA	/LFLAG SET?
0320	0247	5257		JMP LOUT	/YES
0321	0250	1323		TAD LINK	/NO
0322	0251	7010		RAR	/RESTORE LINK
0323	0252	1324		TAD HOLDIN	
0324	0253	3000		DCA 0000	/RESTORE PC
0325	0254	1325		TAD AC	/RESTORE AC
0326	0255	6001		ION	/INTERUPT ON
0327	0256	5400		JMP I 0	/RTN TO PROGRAM
0330	0257	3326	LOUT.	DCA LFLAG	/CLEAR LFLAG
0331	0260	6141		LINC	
0332				LMODE	
0333	0261	0011		CLR	
0334	0262	2323		ADD LINK	
0335	0263	0321		ROR I 1	/RESTORE LINK
0336	0264	2325		ADD AC	/RESTORE AC
0337	0265	0006		DJR	
0340	0266	0500		IOB	
0341	0267	6001		ION	/INTERUPT ON
0342	0270	0000	LRTN.	0	
0343				Pmode	
0344			/		
0345			/		
0346			/		
0347			/		
0350			/	SENSE SWITCH ROUTINE	
0351			/		
0352			/	SENSE SW 1 = DUMP MEMORY & RESTART	
0353			/	SENSE SW 2 = DUMP MEMORY & GO TO DIAL	
0354			/		
0355			/		
0356	0271	0000	SENSE.	0	
0357	0272	6141		LINC	
0360				LMODE	
0361	0273	0461		SNS I 1	/SENSE SW 1 SET?
0362	0274	6301		JMP SW1	/YES
0363	0275	0462		SNS I 2	/NO SW 2 SET?
0364	0276	6306		JMP SW2	/YES
0365	0277	0002		PDP	/NO
0366				Pmode	
0367	0300	5671		JMP I SENSE	
0370				LMODE	
0371	0301	0002	SW1.	PDP	
0372				Pmode	
-					
0373	0302	7200		CLA	
0374	0303	7001		IAC	
0375	0304	3036		DCA EFLAG1	/SET SW1 FLAG
0376	0305	5312		JMP EMPTY	
0377				LMODE	
0400	0306	0002	SW2.	PDP	

0401				PMODE	
0402	0307	7300		CLR CLL	
0403	0310	7001		IAC	
0404	0311	3037		DCA EFLAG2	/SET SW2 FLAG
0405	0312	7100	EMPTY,	CLL	
0406	0313	1323		TAD LINK	
0407	0314	7010		RAR	/RESTORE LINK
0410	0315	1324		TAD HOLDIN	
0411	0316	3000		DCA 0000	/RESTORE PC
0412	0317	1325		TAD AC	/RESTORE AC
0413	0320	6244		RMF	/RESTORE IF & OF
0414	0321	5400		JMP I 0	
0415			/		
0416			/		
0417	0322	0020	CFLAG,	0020	
0420	0323	0000	LINK,	0	
0421	0324	0000	HOLDIN,	0	
0422	0325	0000	AC,	0	
0423	0326	0000	LFLAG,	0	
0424			/		
0425			/		
0426			/		
0427			/	AD SAMPLE & DISPLAY ROUTINE	
0430			/		
0431			/		
0432				*460	
0433	0400	0000	SADIS,	0	
0434	0401	6141		LINC	
0435				LMODE	
0436	0402	0011		CLR	
0437	0403	2001		ADD 0001	
0440	0404	4526		STC HOLD	/SAVE PI JMP
0441	0405	0011	AGNDS,	CLR	
0442	0406	0112		SAM 12	/SMPL INPUT SGNL
0443	0407	1560		BCL I	
0444	0410	7000		7000	
0445	0411	4530		STC SIG	/STORE INPUT
0446	0412	2137		ADD THRESH	
0447	0413	2122		ADD M2	
0450	0414	0017		COM	
0451	0415	2530		ADD SIG	
0452	0416	0451		APD	/PULSE OR SPACE?
0453	0417	0467		SKP	/SPACE
0454	0420	6425		JMP P3	/PULSE
0455	0421	0011		CLR	
0456	0422	2077		ADD M1	
0457	0423	1071		STA I SIGNAL	/STORE -1
0460	0424	6430		JMP CONDIS	
0461	0425	0011	P3,	CLR	
0462	0426	2100		ADD K1	
0463	0427	1071		STA I SIGNAL	/STORE +1
0464	0430	0011	CONDIS,	CLR	
0465	0431	0101		SAM 1	/GET SAMPLE NBR
0466	0432	4022		STC COUNT	

0467	0433	0100	SAM 0	/SMPL THRESHOLD
0470	0434	4137	STC THRESH	
0471	0435	2531	ADD M377	
0472	0436	2137	ADD THRESH	
0473	0437	0166	DIS I 6	/DSPLY THRESHOLD
0474	0440	0011	CLR	
0475	0441	2531	ADD M377	
0476	0442	2530	ADD SIG	
0477	0443	0167	DIS I 7	/DSPLY INPUT
0500	0444	0011	CLR	
0501	0445	4001	STC 0001	/CLEAR 0001
0502	0446	2022	ADD COUNT	
0503	0447	1560	BCL I	
0504	0450	0777	0777	
0505	0451	0311	ROR 9	
0506	0452	4033	STC NUMR	/GET FIRST BIT
0507	0453	7001	JMP DISCPY	/DISPLAY IT
0510	0454	2022	ADD COUNT	
0511	0455	1560	BCL I	
0512	0456	7077	7077	
0513	0457	0306	ROR 6	
0514	0460	4033	STC NUMR	/GET SECOND BIT
0515	0461	7001	JMP DISCPY	/DISPLAY IT
0516	0462	2022	ADD COUNT	
0517	0463	1560	BCL I	
0520	0464	7707	7707	
0521	0465	0303	ROR 3	
0522	0466	4033	STC NUMR	/GET THIRD BIT
0523	0467	7001	JMP DISCPY	/DISPLAY IT
0524	0470	2022	ADD COUNT	
0525	0471	1560	BCL I	
0526	0472	7770	7770	
0527	0473	4033	STC NUMR	/GET FOURTH BIT
0530	0474	7001	JMP DISCPY	/DISPLAY IT
0531	0475	0500	IOB	
0532	0476	6031	KSF	/KEYBOARD HIT?
0533	0477	6405	JMP AGNDS	/NO
0534	0500	0500	IOB	/YES
0535	0501	6032	KCC	
0536	0502	2527	ADD MAIND0	
0537	0503	1620	BSE I	
0540	0504	6000	6000	/CREATE JMP
0541	0505	4430	STC COND15	
0542	0506	2022	ADD COUNT	
0543	0507	1660	BCL I	
0544	0510	7777	7777	
0545	0511	2100	ADD K1	/FORM 2S COMP
0546	0512	4131	STC KCHANG	/OF COUNT
0547	0513	2131	ADD KCHANG	
0550	0514	4027	STC CHANGE	
0551	0515	2027	ADD CHANGE	

0552	0516	2027		ADD CHANGE	
0553	0517	2123		ADD KM2	
0554	0520	4020		STC CHECK	/TWICE KCHANG
0555	0521	0011	MAINDS.	CLR	
0556	0522	2526		ADD HOLD	
0557	0523	4001		STC 0001	/RESTORE P1
0560	0524	0002		PDP	/JMP INSTR
0561				Pmode	
0562	0525	5600		JMP I SADS	
0563			/		
0564			/		
0565	0526	0000	HOLD.	0	
0566	0527	0521	MAIND0.	MAINDS	
0567	0530	0000	SIG.	0	
0570	0531	7401	M377.	-377	

0571			/		
0572			/		
0573			/		
0574			/		
0575			/	CHARACTER DISPLAY ROUTINE	
0576			/		
0577			/		
0600				*1000	
0601	1000	0000		0	/SENSE SW1 STORE
0602				Lmode	
0603	1001	0011	DISCPY.	CLR	
0604	1002	2000		ADD 0000	
0605	1003	5114		STC HOLDRN	/SAVE JMS RTN
0606	1004	2033		ADD NUMR	
0607	1005	0470		AZE I	/NUMR = 0?
0610	1006	7061		JMP NUM0	/YES
0611	1007	2123		ADD KM2	/NO
0612	1010	0470		AZE I	/NUMR = 1?
0613	1011	7067		JMP NUM1	/YES
0614	1012	2123		ADD KM2	/NO
0615	1013	0470		AZE I	/NUMR = 2?
0616	1014	7075		JMP NUM2	/YES
0617	1015	2123		ADD KM2	/NO
0620	1016	0470		AZE I	/NUMR = 3?
0621	1017	7103		JMP NUM3	/YES
0622	1020	2123		ADD KM2	/NO
0623	1021	0470		AZE I	/NUMR = 4?
0624	1022	7037		JMP NUM4	/YES
0625	1023	2123		ADD KM2	/NO
0626	1024	0470		AZE I	/NUMR = 5?
0627	1025	7045		JMP NUM5	/YES
0630	1026	2123		ADD KM2	/NO
0631	1027	0470		AZE I	/NUMR = 6?
0632	1030	7053		JMP NUM6	/YES
0633	1031	0011		CLR	/NO NUM = 7
0634	1032	1760		DSC I	

0635	1033	4443		4443	
0636	1034	1760		DSC I	
0637	1035	6050		6050	/DISPLAY 7
0640	1036	7110		JMP DISOUT	
0641	1037	0011	NUM4.	CLR	
0642	1040	1760		DSC I	
0643	1041	2414		2414	
0644	1042	1760		DSC I	
0645	1043	0477		0477	/DISPLAY 4
0646	1044	7110		JMP DISOUT	
0647	1045	0011	NUM5.	CLR	
0650	1046	1760		DSC I	
0651	1047	5172		5172	
0652	1050	1760		DSC I	
0653	1051	0651		0651	/DISPLAY 5
0654	1052	7110		JMP DISOUT	
0655	1053	0011	NUM6.	CLR	
0656	1054	1760		DSC I	
0657	1055	1506		1506	
0660	1056	1760		DSC I	
0661	1057	4225		4225	/DISPLAY 6
0662	1060	7110		JMP DISOUT	
0663	1061	0011	NUM0.	CLR	
0664	1062	1760		DSC I	
0665	1063	4136		4136	
0666	1064	1760		DSC I	
0667	1065	3641		3641	/DISPLAY 0

0670	1066	7110		JMP DISOUT	
0671	1067	0011	NUM1.	CLR	
0672	1070	1760		DSC I	
0673	1071	2101		2101	
0674	1072	1760		DSC I	
0675	1073	0177		0177	/DISPLAY 1
0676	1074	7110		JMP DISOUT	
0677	1075	0011	NUM2.	CLR	
0700	1076	1760		DSC I	
0701	1077	4523		4523	
0702	1100	1760		DSC I	
0703	1101	2151		2151	/DISPLAY 2
0704	1102	7110		JMP DISOUT	
0705	1103	0011	NUM3.	CLR	
0706	1104	1760		DSC I	
0707	1105	4122		4122	
0710	1106	1760		DSC I	
0711	1107	2651		2651	/DISPLAY 3
0712	1110	0011	DISOUT.	CLR	
0713	1111	3114		ADD HOLDEN	
0714	1112	4000		STC 0000	/RESTORE RTN
0715	1113	6000		JMP 0	
0716					
0717					

0720	1114	0000	HOLDRN.	0	
0721					PMODE
0722			/		
0723			/		
0724			/		
0725			/		
0726			/		AD SAMPLE DECISION ROUTINE
0727			/		
0730			/		
0731				*1200	
0732	1200	0000	ADSAM.	0	
0733	1201	4423		JMS I SADI50	/SMPL & DSPLY
0734	1202	7200		CLA	
0735	1203	2027		ISZ CHANGE	/N SAMPLES YET?
0736	1204	5600		JMP I ADSAM	/NO
0737	1205	1146		TAD WHICH	/YES
0740	1206	7440		SZA	/CHANGE OR CHK?
0741	1207	5246		JMP ENDCHK	/END OF CHECK
0742	1210	1127		TAD K6000	/END OF CHANGE
0743	1211	3011		DCA SIGNAL	/RESET MEMORY
0744	1212	1131		TAD KCHANG	
0745	1213	3027		DCA CHANGE	/RESET SMPL NBR
0746	1214	1411		TAD I SIGNAL	
0747	1215	2027		ISZ CHANGE	
0750	1216	5214		JMP -2	/ADD SMPLD INPTS
0751	1217	7700		SMA CLA	/PULSE OR SPACE?
0752	1220	5224		JMP P1	/PULSE
0753	1221	1077		TAD M1	/SPACE
0754	1222	3144		DCA NEW1	/STORE -1
0755	1223	5227		JMP COMP	
0756	1224	7200	P1.	CLA	
0757	1225	7001		IAC	
0760	1226	3144		DCA NEW1	/STORE +1
0761	1227	1127	COMP.	TAD K6000	
0762	1230	3011		DCA SIGNAL	/RESET MEMORY
0763	1231	1136		TAD PAST	/COMPARE WITH
0764	1232	1144		TAD NEW1	/PREVIOUS VALUE
0765	1233	7650		SNA CLA	/CHANGE?
0766	1234	5241		JMP +5	/YES
0767	1235	1131		TAD KCHANG	/Y
0770	1236	3027		DCA CHANGE	/RESET CHANGE
0771	1237	4430		JMS I SENSE0	/CHK SNS SWS
0772	1240	5000		JMP I ADSAM	/RTN TO PROG
0773	1241	1020		TAD CHECK	
0774	1242	3027		DCA CHANGE	/SET SMPL NBR
0775	1243	1100		TAD K1	
0776	1244	3146		DCA WHICH	/SET FLAG
0777	1245	5600		JMP I ADSAM	
1000	1246	7200	ENDCHK.	CLA	
1001	1247	1127		TAD K6000	
1002	1250	3011		DCA SIGNAL	/RESET MEMORY

1003	1251	1020		TAD CHECK	
1004	1252	3027		DCA CHANGE	/RESET SMPL NBR
1005	1253	1411		TAD I SIGNAL	
1006	1254	2027		ISZ CHANGE	
1007	1255	5253		JMP -2	/ADD SMPLD INPTS
1010	1256	7700		SMA CLA	/PULSE OR SPACE?
1011	1257	5263		JMP P2	/PULSE
1012	1260	1077		TAD M1	/SPACE
1013	1261	3144		DCA NEW1	/STORE -1
1014	1262	5266		JMP COMP1	
1015	1263	7200	P2.	CLA	
1016	1264	1100		TAD K1	
1017	1265	3144		DCA NEW1	/STORE +1
1020	1266	1127	COMP1.	TAD K6000	
1021	1267	3011		DCA SIGNAL	/RESET MEMORY
1022	1270	1136		TAD PAST	
1023	1271	1144		TAD NEW1	
1024	1272	7440		SZA	/PERMANENT CHG?
1025	1273	5540		JMP I R0UT0	/NO
1026	1274	1136		TAD PAST	/YES
1027	1275	3145		DCA PORS	/SET P/S FLAG
1030	1276	1144		TAD NEW1	
1031	1277	3136		DCA PAST	/SET NEW LEVEL
1032	1300	6141		LINC	
1033				LMODE	
1034	1301	0101		SAM 1	/SAMPLE COUNT
1035	1302	4022		STC COUNT	
1036	1303	0100		SAM 0	/SMPL THRESHOLD
1037	1304	4137		STC THRESH	
1040	1305	0104		SAM 4	/SAMPLE FUDGE
1041	1306	4107		STC FUDGE	
1042	1307	2107		ADD FUDGE	
1043	1310	2107		ADD FUDGE	
1044	1311	2107		ADD FUDGE	
1045	1312	4107		STC FUDGE	
1046	1313	0002		PDP	
1047				PMODE	
1050	1314	1022		TAD COUNT	/GET LATEST
1051	1315	7041		CIA	/SAMPLE NBR
1052	1316	3131		DCA KCHANG	
1053	1317	1131		TAD KCHANG	
1054	1320	3027		DCA CHANGE	/SET CHANGE NBR
1055	1321	1027		TAD CHANGE	
1056	1322	1027		TAD CHANGE	
1057	1323	3020		DCA CHECK	/SET CHECK NBR
1060	1324	6135		CLSA	
1061	1325	7700		SMA CLA	/CLOCK OVRFLW?
1062	1326	5542		JMP I TIMER0	/NO GET TIME
1063	1327	7240		STA	/YES
1064	1330	3147		DCA TIME	/SET TIME=7777
1065	1331	5541		JMP I RESET0	

1066	1332	5600	ADSAM2.	JMP I ADSAM	
1067			/		
1070			/		
1071				*1400	
1072	1400	7200	TIMER.	CLA	
1073	1401	6133		CLAB	
1074	1402	6137		CLCA	
1075	1403	3147		DCA TIME	/STORE TIME
1076	1404	6133	RESET.	CLAB	
1077	1405	1117		TAD RATE0	
1100	1406	6132		CLLR	/CLEAR CLK CNTR
1101	1407	6135		CLSA	
1102	1410	7200		CLA	
1103	1411	1127		TAD K6000	
1104	1412	6132		CLLR	/RESET CLOCK
1105			/		
1106			/		
1107			/	PREPARE TIME & TYPE OF SIGNAL	
1110			/	FOR STORAGE	
1111			/		
1112			/		
1113	1413	7200		CLA	
1114	1414	1145		TAD PORS	
1115	1415	7700		SMA CLA	/PULSE OR SPACE?
1116	1416	5225		JMP PULSE	/PULSE
1117	1417	1147		TAD TIME	/SPACE
1120	1420	7010		RAR	
1121	1421	7100		CLL	
1122	1422	7004		RAL	/SET AC 11=0
1123	1423	3412		DCA I STORE	/STORE DATA
1124	1424	5233		JMP ADUT1	
1125	1425	7200	PULSE.	CLA	
1126	1426	1147		TAD TIME	
1127	1427	7010		RAR	
1130	1430	7120		STL	
1131	1431	7004		RAL	/SET AC 11=1
1132	1432	3412		DCA I STORE	/STORE DATA
1133	1433	1012	ADUT1.	TAD STORE	
1134	1434	1031		TAD M777	
1135	1435	7440		SZA	/LAST WD OF MEM?
1136	1436	5243		JMP .+5	/NO
1137	1437	1032		TAD K577	/YES
1140	1440	3012		DCA STORE	/RESET MEM LOC
1141	1441	7001		IAC	
1142	1442	3043		DCA MSTART	/SET FLAG
1143	1443	7200	ADUT.	CLA	
1144	1444	3146		DCA WHICH	/RESET CHG FLAG
1145	1445	1131		TAD KCHANG	
1146	1446	3027		DCA CHANGE	/RESET CHANGE
1147	1447	5543		JMP I ADSAM1	/IND RETURN
1150			/		
1151			/		
1152			/		
1153				*1500	

1154	1500	6141	FIG1.	LINC	
1155				LMODE	
1156	1501	0011	MNAGN.	CLR	
1157	1502	2163		ADD TEMPMN	
1160	1503	0241		ROL 1	
1161	1504	4163		STC TEMPMN	
1162	1505	2164		ADD MNUM	
1163	1506	1120		ADA I	
1164	1507	0001		0001	
1165	1510	4164		STC MNUM	
1166	1511	2164		ADD MNUM	
1167	1512	0450		AZE	/MNUM = 0?
1170	1513	7501		JMP MNAGN	/NO
1171	1514	0002		PDP	/YES
1172				Pmode	
1173	1515	5567		JMP I RFIG10	
1174	1516	6141	FIG2.	LINC	
1175				LMODE	
1176	1517	0011	ROTAGN.	CLR	
1177	1520	2163		ADD TEMPMN	
1200	1521	0301		ROR 1	
1201	1522	4163		STC TEMPMN	
1202	1523	2165		ADD LMNUM	
1203	1524	1120		ADA I	
1204	1525	0001		0001	
1205	1526	4165		STC LMNUM	
1206	1527	2165		ADD LMNUM	
1207	1530	0451		APD	/LMNUM POSITIVE?
1210	1531	7517		JMP ROTAGN	/NO
1211	1532	0011		CLR	/YES
1212	1533	2163		ADD TEMPMN	
1213	1534	1560		BCL I	
1214	1535	0077		0077	
1215	1536	2153		ADD NUM32	
1216	1537	4177		STC CDE32	2ND CODE WORD
1217	1540	2163		ADD TEMPMN	
1220	1541	1560		BCL I	
1221	1542	7700		7700	
1222	1543	4163		STC TEMPMN	
1223	1544	0002		PDP	
1224				Pmode	
1225	1545	5571		JMP I RFIG20	
1226			/		
1227			/		
1230			/		
1231			/		
1232			/		
1233			/	MAIN DATA PROCESSING ROUTINE	
1234			/		
1235			/		
1236				*1600	

1237	1600	7300	PROCES, CLA CLL	/
1240	1601	4445	JMS I IFROCD	/INITIALIZE
1241	1602	7200	PRCAGN, CLA	
1242	1603	1013	TAD WORD	
1243	1604	7041	CIA	
1244	1605	1012	TAD STORE	
1245	1606	7540	SMA SZA	/STORE-WORD=0?
1246	1607	5223	JMP GETNW	/YES
1247	1610	7200	CLA	/NO
1250	1611	1043	TAD MSTART	
1251	1612	7440	SZA	/MSTART SET?
1252	1613	5223	JMP GETNW	/YES
1253	1614	1036	TAD EFLAG1	/NO
1254	1615	7440	SZA	/EFLAG1 SET?
1255	1616	5434	JMP I EFL10	/YES
1256	1617	1037	TAD EFLAG2	/NO
1257	1620	7440	SZA	/EFLAG2 SET?
1260	1621	5435	JMP I EFL20	/YES
1261	1622	5202	JMP PRCAGN	/NO
1262	1623	7300	GETNW, CLA CLL	
1263	1624	1413	TAD I WORD	

1264	1625	3111	DCA HOLDWD	
1265	1626	1111	TAD HOLDWD	
1266	1627	3414	DCA I STORIT	/STORE TIME
1267	1630	1014	TAD STORIT	
1270	1631	1321	TAD M4050	
1271	1632	7640	SZA CLA	/STORIT = 4050?
1272	1633	5236	JMP +3	/NO
1273	1634	1154	TAD KSTOR	/YES
1274	1635	3014	DCA STORIT	/RESET STORIT
1275	1636	1013	TAD WORD	
1276	1637	1031	TAD M777	
1277	1640	7440	SZA	/WORD=777?
1300	1641	5245	JMP PORSWD	/NO
1301	1642	1032	TAD K577	/YES
1302	1643	3013	DCA WORD	/RESET WORD
1303	1644	3043	DCA MSTART	/CLEAR MSTART
1304	1645	7300	PORSWD, CLA CLL	
1305	1646	1111	TAD HOLDWD	
1306	1647	7010	RAR	
1307	1650	7430	SZL	/PULSE OR SPACE?
1310	1651	5524	JMP I PWD0	/PULSE
1311	1652	7200	CLA	/SPACE
1312	1653	1102	TAD PX	
1313	1654	7440	SZA	/LAST PULSE?
1314	1655	5533	JMP I SDASH0	/DASH
1315	1656	1111	TAD HOLDWD	/DOT
1316	1657	7010	RAR	/COMPARE WITH
1317	1660	3113	DCA TEMP	/AVERAGE
1320	1661	1106	TAD PAVG	
1321	1662	7010	RAR	

1322	1663	7041	CIA	
1323	1664	1113	TAD TEMP	
1324	1665	7710	SPA CLA	/INTER OR INTRA?
1325	1666	5473	JMP I PRCPG0	/INTRA
1326	1667	1120	TAD CWAVG	/INTER
1327	1670	7012	RTR	/COMPUTE NEW
1330	1671	7010	RAR	/CWAVG
1331	1672	0135	AND K777	
1332	1673	7041	CIA	
1333	1674	3322	DCA CTEMP	
1334	1675	1111	TAD HOLDWD	
1335	1676	7012	RTR	
1336	1677	7010	RAR	
1337	1700	0135	AND K777	
1340	1701	1322	TAD CTEMP	
1341	1702	1120	TAD CWAVG	
1342	1703	3120	DCA CWAVG	/NEW CWAVG
1343	1704	1120	TAD CWAVG	
1344	1705	1107	TAD FUDGE	/ADD CORRECTION
1345	1706	3110	DCA SAFUDG	/NEW SAFUDG
1346	1707	7100	CLL	
1347	1710	1110	TAD SAFUDG	
1350	1711	7010	RAR	
1351	1712	7041	CIA	
1352	1713	1113	TAD TEMP	
1353	1714	7710	SPA CLA	/INTER CH OR WD?
1354	1715	5525	JMP I ICODE0	/INTER CHAR
1355	1716	1111	TAD HOLDWD	/INTER WORD
1356	1717	3076	DCA IWFLAG	
1357	1720	5525	JMP I ICODE0	
1360				
1361	1721	3730	M4050, -4050	
1362	1722	0000	CTEMP, 0	
1363				
1364				
1365	1723	7300	CALSPA, CLA CLL	/COMPUTE
1366	1724	1120	TAD CWAVG	/INITIAL
1367	1725	7012	RTR	/CWAVG
1370	1726	0101	AND K1777	
1371	1727	7041	CIA	
1372	1730	3113	DCA TEMP	
1373	1731	1111	TAD HOLDWD	
1374	1732	7012	RTR	
1375	1733	0101	AND K1777	
1376	1734	1113	TAD TEMP	
1377	1735	1120	TAD CWAVG	
1400	1736	3120	DCA CWAVG	/NEW CWAVG
1401	1737	5575	JMP I IPRTN0	
1402				
1403				
1404				

1405				*2000	
1406	2000	7300	SDASH.	CLA CLL	
1407	2001	1106		TAD PAVG	
1410	2002	7041		CIA	
1411	2003	1102		TAD PX	/COMPUTE
1412	2004	7012		RTR	/INTER/INTRA
1413	2005	0101		AND K1777	/SPACE(DASH)
1414	2006	7041		CIA	/EQUATION
1415	2007	1106		TAD PAVG	
1416	2010	7100		CLL	
1417	2011	7010		RAR	
1420	2012	7041		CIA	
1421	2013	3113		DCA TEMP	
1422	2014	1111		TAD HOLDWD	
1423	2015	7100		CLL	
1424	2016	7010		RAR	
1425	2017	1113		TAD TEMP	
1426	2020	7700		SMA CLA	/INTER OR INTRA?
1427	2021	5223		JMP +2	/INTER
1430	2022	5473		JMP I PRCAG0	/INTRA
1431	2023	7300		CLA CLL	
1432	2024	1106		TAD PAVG	
1433	2025	7041		CIA	/COMPUTE
1434	2026	1102		TAD PX	/INTER CH
1435	2027	7012		RTR	/OR
1436	2030	0101		AND K1777	/INTER WORD
1437	2031	7041		CIA	/EQUATION
1440	2032	1110		TAD SAFUDG	
1441	2033	7100		CLL	
1442	2034	7010		RAR	
1443	2035	7041		CIA	
1444	2036	3113		DCA TEMP	
1445	2037	1111		TAD HOLDWD	
1446	2040	7100		CLL	
1447	2041	7010		RAR	
1450	2042	1113		TAD TEMP	
1451	2043	7710		SPA CLA	/CH OR WORD?
1452	2044	5525		JMP I IC0DE0	/INTER CH
1453	2045	1111		TAD HOLDWD	/INTER WORD
1454	2046	3076		DCA INFLAG	/SET INFLAG
1455	2047	5525		JMP I IC0DE0	
1456	2050	7200	PWD.	CLA	
1457	2051	1021		TAD NUM	
1460	2052	7001		IAC	
1461	2053	3021		DCA NUM	/INCRMNT NUM
-					
1462	2054	1111		TAD HOLDWD	
1463	2055	7041		CIA	
1464	2056	1106		TAD PAVG	
1465	2057	7700		SMA CLA	/DOT OR DASH?
1466	2060	5320		JMP PDOT	/DOT
1467	2061	7100		CLL	/DASH

1470	2062	1111		TAD HOLDWD	
1471	2063	7010		RAR	
1472	2064	7041		CIA	
1473	2065	1105		TAD DASHAV	
1474	2066	7700		SMA CLA	/HOLDWD>2DASHAV?
1475	2067	5274		JMP .+5	/NO
1476	2070	1105		TAD DASHAV	/YES
1477	2071	1105		TAD DASHAV	
1500	2072	1100		TAD K1	
1501	2073	3111		DCA HOLDWD	/HOLDWD=2DASHAV
1502	2074	1111		TAD HOLDWD	
1503	2075	3102		DCA PX	/STORE IN PX
1504	2076	7120		STL	
1505	2077	1044		TAD WOREG	
1506	2100	7004		RAL	
1507	2101	3044		DCA WOREG	/PUT 1 IN WOREG
1510	2102	1111		TAD HOLDWD	
1511	2103	7012		RTR	
1512	2104	7010		RAR	
1513	2105	0135		AND K777	/COMPUTE
1514	2106	3113		DCA TEMP	/NEW
1515	2107	1105		TAD DASHAV	/DASHAV
1516	2110	7012		RTR	/AVERAGE
1517	2111	7010		RAR	
1520	2112	0135		AND K777	
1521	2113	7041		CIA	
1522	2114	1113		TAD TEMP	
1523	2115	1105		TAD DASHAV	
1524	2116	3105		DCA DASHAV	/NEW AVERAGE
1525	2117	5343		JMP PAVGCP	
1526	2120	7200	PDOT.	CLA	
1527	2121	3102		DCA PX	/CLEAR PX
1530	2122	7100		CLL	
1531	2123	1044		TAD WOREG	
1532	2124	7004		RAL	
1533	2125	3044		DCA WOREG	/PUT 0 IN WOREG
1534	2126	1111		TAD HOLDWD	
1535	2127	7012		RTR	
1536	2130	7010		RAR	
1537	2131	0135		AND K777	
1540	2132	3113		DCA TEMP	/DOTAV
1541	2133	1104		TAD DOTAV	/AVERAGE
1542	2134	7012		RTR	
1543	2135	7010		RAR	
1544	2136	0135		AND K777	
1545	2137	7041		CIA	
1546	2140	1113		TAD TEMP	
1547	2141	1104		TAD DOTAV	
1550	2142	3104		DCA DOTAV	/NEW AVERAGE
1551	2143	7100	PAVGCP.	CLL	/COMPUTE
1552	2144	1104		TAD DOTAV	
1553	2145	7010		RAR	
1554	2146	7100		CLL	
1555	2147	1105		TAD DASHAV	/NEW PAVG

1556	2150	7010		RAR	
1557	2151	3106		DCA PAVG	/NEW AVERAGE
1560	2152	5473		JMP I PROCAG0	
1561					
1562					
1563				*2200	
1564	2200	7200	ICODE,	CLA	
1565	2201	1021		TAD NUM	/COMPUTE
1566	2202	7041		CIA	/INTERNAL
1567	2203	1121		TAD K13	/CODE
1570	2204	3210		DCA SHFTN	/WORD
1571	2205	7621		CAM	
1572	2206	1044		TAD WDREG	
1573	2207	7413		SHL	
1574	2210	0000	SHFTN,	0	
1575	2211	1021		TAD NUM	
1576	2212	3046		DCA CODE	/STORE IT
1577	2213	5472		JMP I ASC0	
1600	2214	7200	EFL1,	CLA	
1601	2215	2122		ISZ M2	/FIRST TIME?
1602	2216	5246		JMP EFL11	/YES
1603	2217	7200		CLA	/NO SECOND
1604	2220	3036		DCA EFLAG1	/RESET EFLAG1
1605	2221	1077		TAD M1	
1606	2222	3136		DCA PAST	/RESET PAST
1607	2223	1032		TAD K577	
1610	2224	3013		DCA WORD	/RESET WORD
1611	2225	1032		TAD K577	
1612	2226	3012		DCA STORE	/RESET STORE
1613	2227	1026		TAD KCLR	
1614	2230	3532		DCA I COND10	/RESET CLR INST
1615	2231	1123		TAD KM2	
1616	2232	3122		DCA M2	/RESET M2
1617	2233	1074		TAD KLINE	
1620	2234	3075		DCA LINE	/RESET LINE
1621	2235	3172		DCA TALLY	/RESET TALLY
1622	2236	3173		DCA TALLY1	
1623	2237	3174		DCA TALLY2	
1624	2240	3120		DCA CWAVG	
1625	2241	3104		DCA DOTAV	
1626	2242	3105		DCA DASHAV	
1627	2243	3106		DCA PAVG	
1630	2244	4300		JMS SETPTR	/CR + 2 LFS
1631	2245	5526		JMP I RSTAR0	/RESTART PROG
1632	2246	1123	EFL11,	TAD KM2	
1633	2247	3412		DCA I STORE	/PUT LW SP IN ST
1634	2250	5473		JMP I PROCAG0	/PRINT LAST CH
1635	2251	7200	EFL2,	CLA	
1636	2252	2122		ISZ M2	/FIRST TIME?
1637	2253	5246		JMP EFL11	/YES
1640	2254	6141	DIAL,	LINC	/NO

1641				LMODE	
1642	0255	0011		CLR	
1643	0256	0001		AXO	/CLEAR FLAGS
1644	0257	0602		LIF 2	
1645	0260	6016		JMP 16	
1646				SEGMNT 2	
1647				*16	
1650	0016	0701		0701	
1651	0017	7300		7300	/RTN TO DIAL
1652				PMODE	
1653			/		
1654			/		
1655			/		
1656				*2300	
1657	2300	0000	SETPTR, 0		
-					
1660	2301	7200		CLA	
1661	2302	1115		TAD K215	
1662	2303	4311		JMS TYPE	/CR
1663	2304	1116		TAD K212	
1664	2305	4311		JMS TYPE	/LF
1665	2306	1116		TAD K212	
1666	2307	4311		JMS TYPE	/LF
1667	2310	5700		JMP I SETPTR	
1670	2311	0000	TYPE, 0		
1671	2312	6041		TSF	
1672	2313	5312		JMP -1	
1673	2314	6046		TLS	/PRINT CH
1674	2315	7200		CLA	
1675	2316	5711		JMP I TYPE	
1676			/		
1677			/		
1700			/		
1701			/		
1702			/	INITIAL DATA PROCESSING ROUTINE	
1703			/		
1704			/		
1705				*2400	
1706	2400	0000	IPROC, 0		
1707	2401	1127		TAD K6000	
1710	2402	3011		DCA SIGNAL	/RESET SIGNAL
1711	2403	6001		ION	/INTERUPT ON
1712	2404	7200	IPRTN, CLA		
1713	2405	1013		TAD WORD	
1714	2406	1376		TAD M741	
1715	2407	7650		SNA CLA	/98 WORDS YET?
1716	2410	5365		JMP IPEXIT	/YES
1717	2411	1012		TAD STORE	/NO
1720	2412	7041		CIA	
1721	2413	1013		TAD WORD	
1722	2414	7640		SZA CLA	/WORD = STORE?
1723	2415	5222		JMP +5	/NO

1724	2416	1037	TAD EFLAG2	/YES
1725	2417	7440	SZA	/EFLAG2 SET?
1726	2420	5534	JMP I DIAL0	/YES
1727	2421	5204	JMP IPRTN	/NO CHK AGN
1730	2422	7200	CLA	
1731	2423	1413	TAD I WORD	/GET NEW WORD
1732	2424	3111	DCA HOLDWD	/STORE IT
1733	2425	1111	TAD HOLDWD	
1734	2426	7100	CLL	
1735	2427	7010	RAR	
1736	2430	7620	SNL CLA	/PULSE OR SPACE?
1737	2431	5370	JMP GT32	/SPACE
1740	2432	1013	TAD WORD	/PULSE
1741	2433	1103	TAD M640	
1742	2434	7710	SPA CLA	/32 WORDS YET?
1743	2435	5315	JMP IPPUL	/NO
1744	2436	7100	CLL	/YES
1745	2437	1111	TAD HOLDWD	
1746	2440	7010	RAR	
1747	2441	7041	CIA	
1750	2442	3113	DCA TEMP	
1751	2443	7100	CLL	
1752	2444	1106	TAD PAVG	
1753	2445	7010	RAR	
1754	2446	1113	TAD TEMP	
1755	2447	7700	SMA CLA	/DOT OR DASH?
1756	2450	5267	JMP IPDOT	/DOT
-				
1757	2451	1105	TAD DASHAV	/DASH
1760	2452	7012	RTR	
1761	2453	7010	RAR	
1762	2454	0135	AND K777	/COMPUTE
1763	2455	7041	CIA	/NEW
1764	2456	3113	DCA TEMP	/DASHAV
1765	2457	1111	TAD HOLDWD	/AVERAGE
1766	2460	7012	RTR	
1767	2461	7010	RAR	
1770	2462	0135	AND K777	
1771	2463	1113	TAD TEMP	
1772	2464	1105	TAD DASHAV	
1773	2465	3105	DCA DASHAV	/NEW AVERAGE
1774	2466	5305	JMP IPPAV	
1775	2467	7300	IPDOT, CLA CLL	
1776	2470	1104	TAD DOTAV	
1777	2471	7012	RTR	
2000	2472	7010	RAR	/COMPUTE
2001	2473	0135	AND K777	/NEW
2002	2474	7041	CIA	/DOTAV
2003	2475	3113	DCA TEMP	/AVERAGE
2004	2476	1111	TAD HOLDWD	
2005	2477	7012	RTR	
2006	2500	7010	RAR	

2007	2501	0135		AND K777	
2010	2502	1113		TAD TEMP	
2011	2503	1104		TAD DOTAV	
2012	2504	3104		DCA DOTAV	/NEW AVERAGE
2013	2505	7100	IPPAV,	CLL	/COMPUTE NEW
2014	2506	1104		TAD DOTAV	/PAVG
2015	2507	7010		RAR	
2016	2510	7100		CLL	
2017	2511	1105		TAD DASHAV	
2020	2512	7010		RAR	
2021	2513	3106		DCA PAVG	/NEW AVERAGE
2022	2514	5204		JMP IPRTN	
2023	2515	7300	IPPUI,	CLA CLL	
2024	2516	1111		TAD HOLDWD	
2025	2517	7010		RAR	
2026	2520	7041		CIA	
2027	2521	3113		DCA TEMP	
2030	2522	7100		CLL	
2031	2523	1106		TAD PAVG	
2032	2524	7010		RAR	
2033	2525	1113		TAD TEMP	
2034	2526	7700		SMA CLA	/DOT OR DASH?
2035	2527	5342		JMP IPPDOT	/DOT
2036	2530	7100		CLL	/DASH
2037	2531	1111		TAD HOLDWD	
2040	2532	7010		RAR	/COMPUTE
2041	2533	3113		DCA TEMP	/NEW
2042	2534	7100		CLL	/DASHAV
2043	2535	1105		TAD DASHAV	/AVERAGE
2044	2536	7010		RAR	
2045	2537	1113		TAD TEMP	
2046	2540	3105		DCA DASHAV	/NEW AVERAGE
2047	2541	5353		JMP IPPPAV	
2050	2542	7300	IPPDOT,	CLA CLL	
2051	2543	1111		TAD HOLDWD	/COMPUTE
2052	2544	7010		RAR	/NEW
2053	2545	3113		DCA TEMP	/DOTAV
2054	2546	7100		CLL	/AVERAGE
2055	2547	1104		TAD DOTAV	
-					
2056	2550	7010		RAR	
2057	2551	1113		TAD TEMP	
2060	2552	3104		DCA DOTAV	/NEW AVERAGE
2061	2553	7100	IPPPAV,	CLL	
2062	2554	1105		TAD DASHAV	
2063	2555	7010		RAR	/COMPUTE NEW
2064	2556	3113		DCA TEMP	/PAVG
2065	2557	7100		CLL	
2066	2560	1104		TAD DOTAV	
2067	2561	7010		RAR	
2070	2562	1113		TAD TEMP	
2071	2563	3106		DCA PAVG	/NEW AVERAGE

2072	2564	5204		JMP IPRTN
2073	2565	1032	IPEXIT.	TAD K577
2074	2566	3013		DCA WORD /RESET WORD
2075	2567	5600		JMP I IPROC
2076	2570	1013	GT32.	TAD WORD
2077	2571	1103		TAD M640
2100	2572	7710		SPA CLA /32 WORDS YET?
2101	2573	5204		JMP IPRTN /NO
2102	2574	5576		JMP I CALSP0 /YES
2103			/	
2104			/	
2105	2575	7142	M636.	-636
2106	2576	7037	M741.	-741
2107			/	
2110			/	
2111			/	
2112			/	
2113			/	INTERNAL TO ASCII CODE PROGRAM
2114			/	
2115			/	
2116				*2600
2117	2600	7000	J1.	7000
2120	2601	6000	J2.	6000
2121	2602	5000	J3.	5000
2122	2603	4000	J4.	4000
2123	2604	3000	J5.	3000
2124	2605	2000	J6.	2000
2125	2606	1000	J7.	1000
2126	2607	0000	ASCII.	0
2127	2610	7200		CLA
2130	2611	1046		TAD CODE
2131	2612	7700		SMA CLA
2132	2613	5234		JMP PCODE
2133	2614	1206		TAD J7
2134	2615	1046		TAD CODE
2135	2616	7700		SMA CLA
2136	2617	5350		JMP C7000
2137	2620	1205		TAD J6
2140	2621	1046		TAD CODE
2141	2622	7700		SMA CLA
2142	2623	5333		JMP C6000
2143	2624	1204		TAD J5
2144	2625	1046		TAD CODE
2145	2626	7700		SMA CLA
2146	2627	5327		JMP C5000
2147	2630	1203		TAD J4
2150	2631	1046		TAD CODE
2151	2632	7700		SMA CLA
2152	2633	5312		JMP C4000
2153	2634	7200	PCODE.	CLA
2154	2635	1202		TAD J3

2155	2636	1046	TAD CODE
2156	2637	7700	SMA CLA
2157	2640	5306	JMP C3000
2160	2641	1201	TAD J2
2161	2642	1046	TAD CODE
2162	2643	7700	SMA CLA
2163	2644	5271	JMP C2000
2164	2645	1200	TAD J1
2165	2646	1046	TAD CODE
2166	2647	7700	SMA CLA
2167	2650	5265	JMP C1000
2170	2651	1465	TAD I CN5
2171	2652	1046	TAD CODE
2172	2653	7510	SPA
2173	2654	5261	JMP +5
2174	2655	7200	CLA
2175	2656	1047	TAD C01
2176	2657	3010	DCA 10
2177	2660	5464	JMP I CHKIT
2200	2661	7200	CLA
2201	2662	1050	TAD C02
2202	2663	3010	DCA 10
2203	2664	5464	JMP I CHKIT
2204	2665	7200 C1000.	CLA
2205	2666	1051	TAD C11
2206	2667	3010	DCA 10
2207	2670	5464	JMP I CHKIT
2210	2671	7200 C2000.	CLA
2211	2672	1466	TAD I CWAIT
2212	2673	1046	TAD CODE
2213	2674	7510	SPA
2214	2675	5302	JMP +5
2215	2676	7200	CLA
2216	2677	1052	TAD C21
2217	2700	3010	DCA 10
2220	2701	5464	JMP I CHKIT
2221	2702	7200	CLA
2222	2703	1053	TAD C22
2223	2704	3010	DCA 10
2224	2705	5464	JMP I CHKIT
2225	2706	7200 C3000.	CLA
2226	2707	1054	TAD C31
2227	2710	3010	DCA 10
2230	2711	5464	JMP I CHKIT
2231	2712	7200 C4000.	CLA
2232	2713	1467	TAD I CN6
2233	2714	1046	TAD CODE
2234	2715	7510	SPA
2235	2716	5323	JMP +5
2236	2717	7200	CLA
2237	2720	1055	TAD C41
2240	2721	3010	DCA 10
2241	2722	5464	JMP I CHKIT
2242	2723	7200	CLA

2243	2724	1056		TAD C42
2244	2725	3010		DCA 10
2245	2726	5464		JMP I CHKIT
2246	2727	7200	C5000,	CLA
2247	2730	1057		TAD C51
2250	2731	3010		DCA 10
2251	2732	5464		JMP I CHKIT
2252	2733	7200	C6000,	CLA
2253	2734	1470		TAD I CN7
-				
2254	2735	1046		TAD CODE
2255	2736	7510		SPA
2256	2737	5344		JMP +5
2257	2740	7200		CLA
2260	2741	1060		TAD C61
2261	2742	3010		DCA 10
2262	2743	5464		JMP I CHKIT
2263	2744	7200		CLA
2264	2745	1061		TAD C62
2265	2746	3010		DCA 10
2266	2747	5464		JMP I CHKIT
2267	2750	7200	C7000,	CLA
2270	2751	1471		TAD I CN9
2271	2752	1046		TAD CODE
2272	2753	7510		SPA
2273	2754	5361		JMP +5
2274	2755	7200		CLA
2275	2756	1062		TAD C71
2276	2757	3010		DCA 10
2277	2760	5464		JMP I CHKIT
2300	2761	7200		CLA
2301	2762	1063		TAD C72
2302	2763	3010		DCA 10
2303	2764	5464		JMP I CHKIT
2304			/	
2305			/	
2306			/	
2307			/	
2310				*3000
2311	3000	0275	AUNK,	275
2312	3001	7173	N3,	7173
2313	3002	0263	A3,	263
2314	3003	7272	ENDW,	7272
2315	3004	0244	AENDW,	244
2316	3005	7374	NV,	7374
2317	3006	0326	V,	326
2320	3007	7573	N4,	7573
2321	3010	0264	A4,	264
2322	3011	7770	ERR,	7770
2323	3012	0274	AERR,	274
2324	3013	7773	N5,	7773
2325	3014	0265	A5,	265

2326	3015	7774	NH,	7774
2327	3016	0310	H,	310
2330	3017	7775	NS,	7775
2331	3020	0323	S,	323
2332	3021	7776	NI,	7776
2333	3022	0311	AI,	311
2334	3023	7777	NE,	7777
2335	3024	0305	E,	305
2336	3025	0000	SPAC,	0000
2337	3026	0240	ASPC,	240
2340	3027	6173	N2,	6173
2341	3030	0262	A2,	262
2342	3031	6372	QM,	6372
2343	3032	0277	ADM,	277
2344	3033	6774	NF,	6774
2345	3034	0306	F,	306
2346	3035	6775	NU,	6775
2347	3036	0325	AU,	325
2350	3037	5272	PER,	5272
2351	3040	0256	APER,	256
2352	3041	5373	ENDM,	5373
-				
2353	3042	0252	AENDM,	252
2354	3043	5773	WAIT,	5773
2355	3044	0334	AWAIT,	334
2356	3045	5774	NL,	5774
2357	3046	0314	L,	314
2360	3047	5775	NR,	5775
2361	3050	0322	R,	322
2362	3051	5776	NA,	5776
2363	3052	0301	A,	301
2364	3053	4173	N1,	4173
2365	3054	0261	A1,	261
2366	3055	4374	NJ,	4374
2367	3056	0312	J,	312
2370	3057	4774	NP,	4774
2371	3060	0320	P,	320
2372	3061	4775	NW,	4775
2373	3062	0327	W,	327
2374	3063	3373	FRAC,	3373
2375	3064	0257	AFRAC,	257
2376	3065	3374	NX,	3374
2377	3066	0330	X,	330
2400	3067	3573	DASH,	3573
2401	3070	0255	ADASH,	255
2402	3071	3773	N6,	3773
2403	3072	0266	A6,	266
2404	3073	3774	N8,	3774
2405	3074	0302	B,	302
2406	3075	3775	ND,	3775
2407	3076	0304	D,	304
2410	3077	3776	NN,	3776

2411	3100	0316	N.	316
2412	3101	3777	NT.	3777
2413	3102	0324	T.	324
2414	3103	2272	PAR.	2272
2415	3104	0251	APAR.	251
2416	3105	2374	NY.	2374
2417	3106	0331	Y.	331
2420	3107	2572	SCOL.	2572
2421	3110	0273	ASCOL.	273
2422	3111	2774	NC.	2774
2423	3112	0303	C.	303
2424	3113	2775	NK.	2775
2425	3114	0313	K.	313
2426	3115	1374	NQ.	1374
2427	3116	0321	Q.	321
2430	3117	1472	COMM.	1472
2431	3120	0254	ACOMM.	254
2432	3121	1773	N7.	1773
2433	3122	0267	A7.	267
2434	3123	1774	N2.	1774
2435	3124	0332	A2.	332
2436	3125	1775	NG.	1775
2437	3126	0307	G.	307
2440	3127	1776	NM.	1776
2441	3130	0315	M.	315
2442	3131	0173	N0.	0173
2443	3132	0260	A0.	260
2444	3133	0373	N9.	0373
2445	3134	0271	A9.	271
2446	3135	0772	COL.	0772
2447	3136	0272	ACOL.	272
2450	3137	0773	N8.	0773
2451	3140	0270	AS.	270
2452	3141	0775	NO.	0775
2453	3142	0317	O.	317
2454		/		
2455		/		
2456		/		
2457		/		
2460				*3200
2461	3200	7772	UNK.	7772
2462	3201	7772	UNK1.	7772
2463	3202	7200	CHCK.	CLA
2464	3203	1410		TAD I 10
2465	3204	1046		TAD CODE
2466	3205	7450		SNA
2467	3206	5214		JMP . +6
2470	3207	1410		TAD I 10
2471	3210	2201		ISZ UNK1
2472	3211	7410		SKP
2473	3212	5216		JMP UNK2

2474	3213	5202	JMP CHCK	
2475	3214	1410	TAD I 10	
2476	3215	5224	JMP ATSF	
2477	3216	7200	UNK2, CLA	
2500	3217	1172	TAD TALLY	
2501	3220	7001	IAC	
2502	3221	3172	DCA TALLY	
2503	3222	4562	JMS I UNKCH0	
2504	3223	1447	TAD I C01	
2505	3224	6041	ATSF, TSF	
2506	3225	5224	JMP .-1	
2507	3226	6046	TLS	
2510	3227	3350	DCA SPACHK	/STORE ASCII CH
2511	3230	3021	DCA NUM	/CLEAR NUM
2512	3231	1154	TAD KSTOR	
2513	3232	3014	DCA STORIT	/RESET STORIT
2514	3233	3044	DCA WDREG	/CLEAR WDREG
2515	3234	1200	TAD UNK	
2516	3235	3201	DCA UNK1	
2517	3236	1076	TAD IWFLAG	
2520	3237	7650	SNA CLA	/IWFLAG SET?
2521	3240	5323	JMP LNCK	/NO
2522	3241	1076	TAD IWFLAG	/YES
2523	3242	3150	DCA IWFL	/STORE SPACE
2524	3243	3076	DCA IWFLAG	/CLEAR IWFLAG
2525	3244	2075	ISZ LINE	/END OF PTR LNE?
2526	3245	5247	JMP .+2	/NO
2527	3246	5325	JMP RSLN	/YES
2530	3247	1350	TAD SPACHK	
2531	3250	1351	TAD MI	
2532	3251	7650	SNA CLA	/ASCII = I?
2533	3252	5277	JMP CHORWD	/YES
2534	3253	1350	TAD SPACHK	/NO
2535	3254	1352	TAD MJ	
2536	3255	7650	SNA CLA	/ASCII = J?
2537	3256	5277	JMP CHORWD	/YES
2540	3257	1350	TAD SPACHK	/NO
2541	3260	1353	TAD MQ	
2542	3261	7650	SNA CLA	/ASCII = Q?
2543	3262	5277	JMP CHORWD	/YES
2544	3263	1350	TAD SPACHK	/NO
2545	3264	1354	TAD MU	
2546	3265	7650	SNA CLA	/ASCII = U?
2547	3266	5277	JMP CHORWD	/YES
2550	3267	1350	TAD SPACHK	/NO
-				
2551	3270	1355	TAD MV	
2552	3271	7650	SNA CLA	/ASCII = V?
2553	3272	5277	JMP CHORWD	/YES
2554	3273	1350	TAD SPACHK	/NO
2555	3274	1356	TAD MZ	
2556	3275	7640	SZA CLA	/ASCII = Z?

2557	3276	5313		JMP ITSOK	/NO
2560	3277	7100	CHORND.	CLL	/YES
2561	3300	1110		TAD SAFUDG	
2562	3301	1106		TAD PAVG	
2563	3302	7010		RAR	
2564	3303	3113		DCA TEMP	
2565	3304	7100		CLL	
2566	3305	1150		TAD IWF1	
2567	3306	7010		RAR	
2570	3307	7041		CIA	
2571	3310	1113		TAD TEMP	/SPACE > SADDOT
2572	3311	7700		SMA CLA	/ + PAVG?
2573	3312	5330		JMP UNKFLG	/NO OMIT SPACE
2574	3313	2075	ITSOK.	ISZ LINE	/YES. PRNT SPACE
2575	3314	7410		SKP	
2576	3315	5325		JMP RSLN	
2577	3316	1114		TAD K240	
2600	3317	6041		TSF	
2601	3320	5317		JMP -1	
2602	3321	6046		TLS	/PRINT SPACE
2603	3322	5330		JMP UNKFLG	
2604	3323	2075	LNCK.	ISZ LINE	/END OF PTR LINE?
2605	3324	5330		JMP UNKFLG	/NO
2606	3325	4442	RSLN.	JMS I SETPT0	/YES. CR + 2 LFS
2607	3326	1074		TAD KLINE	
2610	3327	3075		DCA LINE	/RESET LINE
2611	3330	7200	UNKFLG.	CLA	
2612	3331	1155		TAD UNFLG	
2613	3332	7650		SMA CLA	/IN ERROR RTNE?
2614	3333	5336		JMP +3	/NO
2615	3334	2161		ISZ ERWDNM	/YES. 2ND WD PTD?
2616	3335	5343		JMP +6	/NO
2617	3336	3155		DCA UNFLG	/YES. RST UNFLG
2620	3337	3156		DCA UNCK1	/RESET
2621	3340	3157		DCA UNCK2	/RESET
2622	3341	3160		DCA UNCK3	/RESET
2623	3342	5473		JMP I PROAG0	/RETURN
2624	3343	1177		TAD CDE32	
2625	3344	3046		DCA CODE	/GET 2ND WORD
2626	3345	1151		TAD IWF2	
2627	3346	3076		DCA IWFLAG	/RESTORE IWFLAG
2630	3347	5472		JMP I ASC0	/PRINT 2ND WORD
2631			/		
2632	3350	0000	SPACHK. 0		
2633	3351	7467	MI.	-311	
2634	3352	7466	MJ.	-312	
2635	3353	7457	MQ.	-321	
2636	3354	7453	MU.	-325	
2637	3355	7452	MV.	-326	
2640	3356	7446	MZ.	-332	
2641			/		
2642			/		
2643			/		
2644			/		

2645	/	ERROR CHECKING ROUTINE		
2646	/			
2647	/			
2650	/	1. ELIMINATE EXTREMELY SMALL DOTS		
2651	/	DUE TO NOISY TRANSMISSION.		
2652	/			
2653	/	2. CORRECT RUN-ON CHARACTERS DUE TO		
2654	/	TOO SMALL AN INTER CHARACTER SPACE		
2655	/	BY DESIGNATING THE LARGEST OF THE		
2656	/	INTRA CHARACTER SPACES AS AN INTER		
2657	/	CHARACTER SPACE.		
2660	/			
2661	/			
2662		*3400		
2663	3400	0000	UNKCHK.	0
2664	3401	7300	CLA CLL	
2665	3402	1156	TAD UNCK1	
2666	3403	7650	SNA CLA	/UNCK1 SET?
2667	3404	5214	JMP UNCK10	/NO. DO CHECK
2670	3405	1157	TAD UNCK2	/YES
2671	3406	7650	SNA CLA	/UNCK2 SET?
2672	3407	5225	JMP UNCK20	/NO. DO CHECK
2673	3410	1160	TAD UNCK3	/YES
2674	3411	7650	SNA CLA	/UNCK3 SET?
2675	3412	5312	JMP UNCK30	/NO. DO CHECK
2676	3413	5600	JMP I UNKCHK	/YES. RETURN
2677	3414	1173	UNCK10. TAD TALLY1	
2700	3415	7001	IAC	
2701	3416	3173	DCA TALLY1	
2702	3417	1021	TAD NUM	
2703	3420	1354	TAD M11	
2704	3421	7700	SMA CLA	/NUM > 8?
2705	3422	5600	JMP I UNKCHK	/YES. PRT ERROR
2706	3423	7001	IAC	
2707	3424	3156	DCA UNCK1	/SET UNCK1 FLAG
2710	3425	1174	UNCK20. TAD TALLY2	
2711	3426	7001	IAC	
2712	3427	3174	DCA TALLY2	
2713	3430	1154	TAD KSTOR	
2714	3431	3014	DCA STORIT	/RESET MEM LOC
2715	3432	1021	TAD NUM	
2716	3433	7041	CIA	
2717	3434	3164	DCA MNUM	
2720	3435	1104	TAD DOTAV	
2721	3436	7012	RTR	
2722	3437	0101	AND K1777	
2723	3440	7041	CIA	
2724	3441	3355	DCA PULCHK	/LOWER DOT TIME
2725	3442	1414	UNCAGN. TAD I STORIT	/GET NEXT PULSE
2726	3443	7100	CLL	
2727	3444	7010	RAR	

2730	3445	1355		TAD PULCHK	
2731	3446	7710		SPA CLA	/PULSE < LIMIT?
2732	3447	5257		JMP ELIM	/YES
2733	3450	2164		ISZ MNUM	/NO. LAST PULSE?
2734	3451	5255		JMP .+4	/NO
2735	3452	7001		IAC	/YES
2736	3453	3157		DCA UNCK2	/SET UNCK2 FLAG
2737	3454	5312		JMP UNCK30	/GO TO NXT CHECK
2740	3455	2014		ISZ STORIT	/INC MEM LOC
2741	3456	5242		JMP UNCAGN	/CHECK NXT PULSE
2742	3457	1164	ELIM.	TAD MNUM	
2743	3460	3113		DCA TEMP	
2744	3461	1044		TAD WDREG	/GET BAD WORD
2745	3462	7010		RAR	
2746	3463	2113		ISZ TEMP	/ERROR IN LINK?

2747	3464	5262		JMP .-2	/NO
2750	3465	3163		DCA TEMPMN	/YES. STORE
2751	3466	5566		JMP I FIG10	
2752	3467	7300	RFIG1.	CLA CLL	
2753	3470	1021		TAD NUM	
2754	3471	1077		TAD M1	
2755	3472	3021		DCA NUM	
2756	3473	1021		TAD NUM	
2757	3474	7041		CIA	
2760	3475	1121		TAD K13	
2761	3476	3302		DCA SHFTN1	
2762	3477	7621		CAM	
2763	3500	1163		TAD TEMPMN	
2764	3501	7413		SHL	
2765	3502	0000	SHFTN1.	0	
2766	3503	1021		TAD NUM	
2767	3504	3046		DCA CODE	/CORRECTED WORD
2770	3505	7001		IAC	
2771	3506	3157		DCA UNCK2	/SET UNCK2 FLAG
2772	3507	7001		IAC	
2773	3510	3160		DCA UNCK3	/SET UNCK3 FLAG
2774	3511	5472		JMP I ASC0	/PRINT NEW WORD
2775	3512	1154	UNCK30.	TAD KSTOR	
2776	3513	1100		TAD K1	
2777	3514	3014		DCA STORIT	/SET MEM TO 1ST
3000	3515	1021		TAD NUM	/SPACE LOCATION
3001	3516	7041		CIA	
3002	3517	1100		TAD K1	
3003	3520	3164		DCA MNUM	
3004	3521	3353		DCA LGSPA	/SET LGSPA = 0
3005	3522	1414	TRYAGN.	TAD I STORIT	/GET SPACE
3006	3523	7100		CLL	
3007	3524	7010		RAR	
3010	3525	3113		DCA TEMP	
3011	3526	1113		TAD TEMP	
3012	3527	1353		TAD LGSPA	

GE/EE/73A-9

3013	3530	7710		SPA CLA	/SPACE > LGSPA?
3014	3531	5337		JMP SMALL	/NO
3015	3532	1113		TAD TEMP	/YES
3016	3533	7041		CIA	
3017	3534	3353		DCA LGSPA	/NEW LGSPA
3020	3535	1164		TAD MNUM	
3021	3536	3165		DCA LMNUM	/LOC OF LGST SPA
3022	3537	2014	SMALL	ISZ STORIT	/INC MEM LOC
3023	3540	2164		ISZ MNUM	/LAST SPACE?
3024	3541	5322		JMP TRYAGN	/NO
3025	3542	1165		TAD LMNUM	/YES
3026	3543	7041		CIA	
3027	3544	3153		DCA NUM32	/LENGTH OF 2ND
3030	3545	1165		TAD LMNUM	/WORD
3031	3546	1021		TAD NIJ	
3032	3547	3152		DCA NUM31	/LENGTH OF 1ST
3033	3550	1044		TAD WOREG	/WORD
3034	3551	3163		DCA TEMPMN	
3035	3552	5570		JMP I FIG20	
3036			/		
3037	3553	0000	LGSPA, 0		
3040	3554	7767	M11, -11		
3041	3555	0000	PULCHK, 0		
3042			/		
3043			/		
3044			/		
3045				*3600	

3046	3600	7200	RFIG2,	CLA	
3047	3601	1152		TAD NUM31	
3050	3602	7041		CIA	
3051	3603	1121		TAD K13	
3052	3604	3210		DCA SHFTN2	
3053	3605	7621		CAM	
3054	3606	1163		TAD TEMPMN	
3055	3607	7413		SHL	
3056	3610	0000	SHFTN2, 0		
3057	3611	1152		TAD NUM31	
3060	3612	3227		DCA CDE31	/1ST CODE WORD
3061	3613	1076		TAD IWFLAG	
3062	3614	3151		DCA INF2	/STORE IWFLAG
3063	3615	3076		DCA IWFLAG	/CLEAR IWFLAG
3064	3616	1122		TAD M2	
3065	3617	3161		DCA ERWDNM	/SET WD COUNTER
3066	3620	7001		IAC	
3067	3621	3160		DCA UNCK3	/SET UNCK3 FLAG
3070	3622	7001		IAC	
3071	3623	3155		DCA UNFLG	/SET UNFLG FLAG
3072	3624	1227		TAD CDE31	
3073	3625	3046		DCA CODE	
3074	3626	5472		JMP I ASC0	/PRNT 1ST WORD
3075			/		
3076	3627	0000	CDE31, 0		

Appendix B

Hand-Sent Morse Code Data Plots

This appendix contains plots of hand-sent Morse code pulse and space time durations. Pulses and spaces are divided into 10 categories each for data analysis purposes. Pulses (DOTs and DASHes) are categorized by their position within a transmitted Morse code character. Spaces are categorized by the type of pulse they follow. Points contained on the individual and combined cluster plots represent time durations of all pulses and following spaces transmitted during a 10-minute period. See Chapter III for further information.

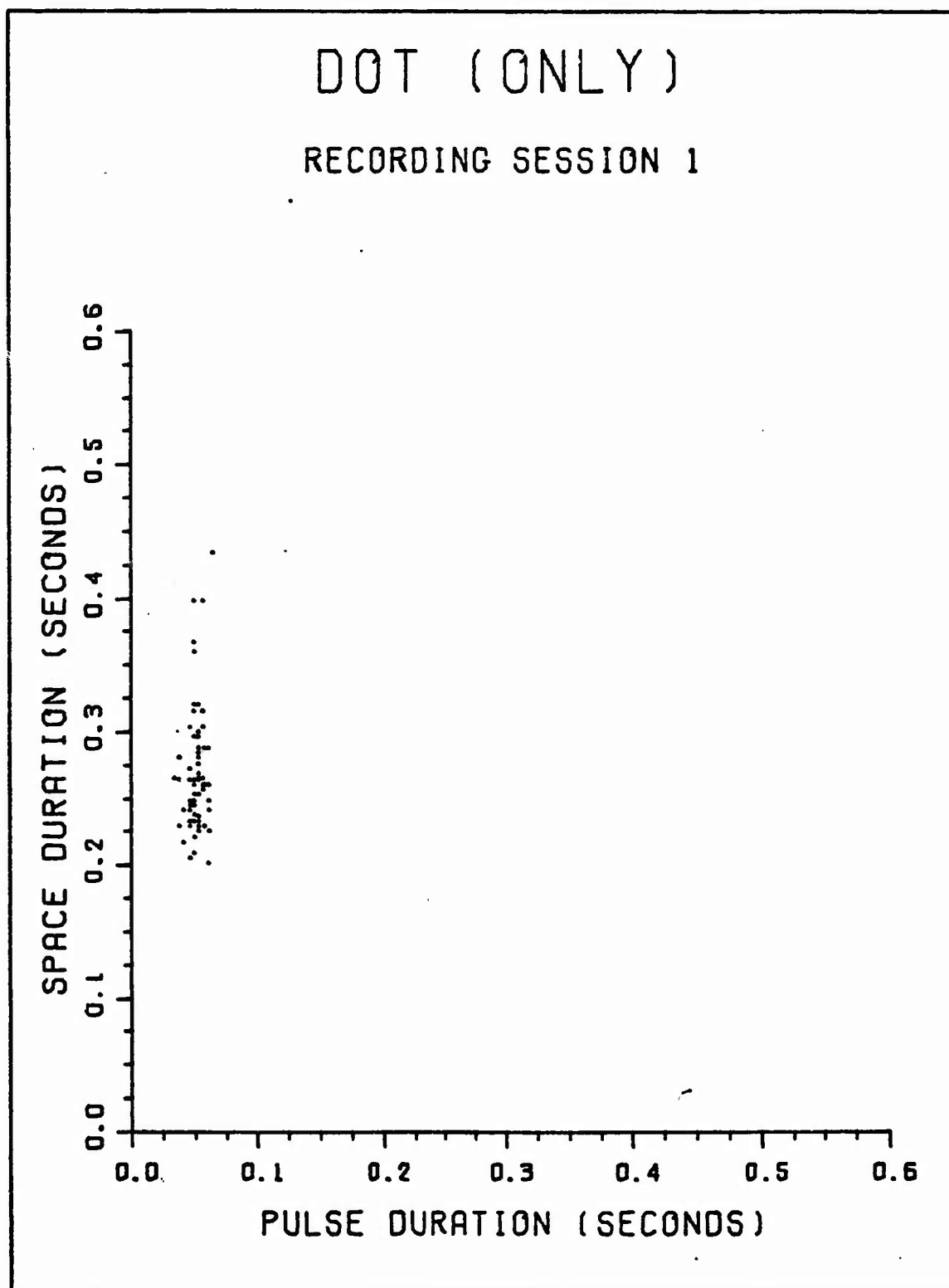


Fig. B-1. Morse Code Data Distribution Plot, DOT (Only)
Time Duration vs. Time Duration of Following Space
(Recording Session 1).

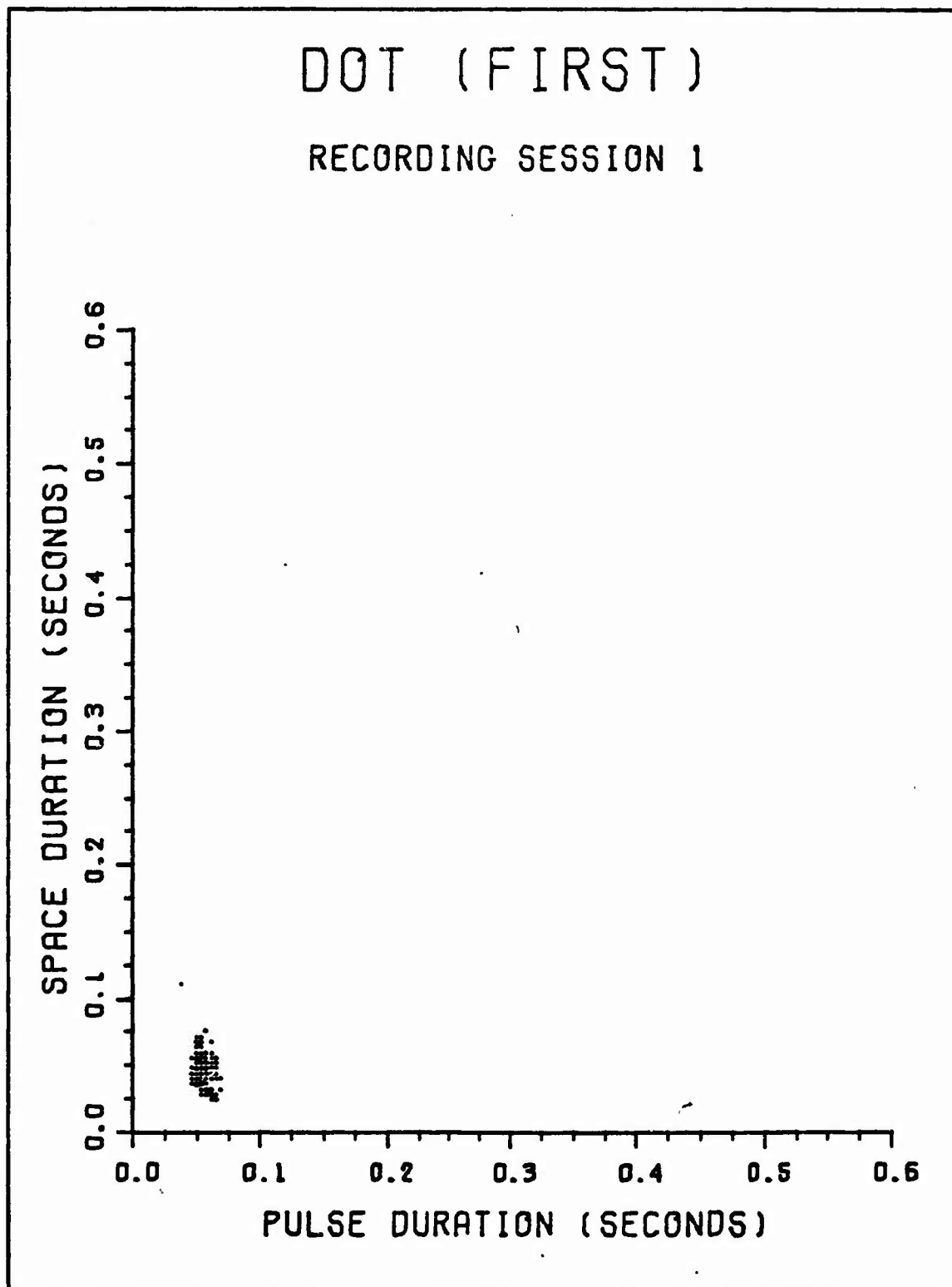


Fig. B-2. Morse Code Data Distribution Plot, DOT (First)
Time Duration vs. Time Duration of Following Space
(Recording Session 1).

DOT (INTERMEDIATE)

RECORDING SESSION 1

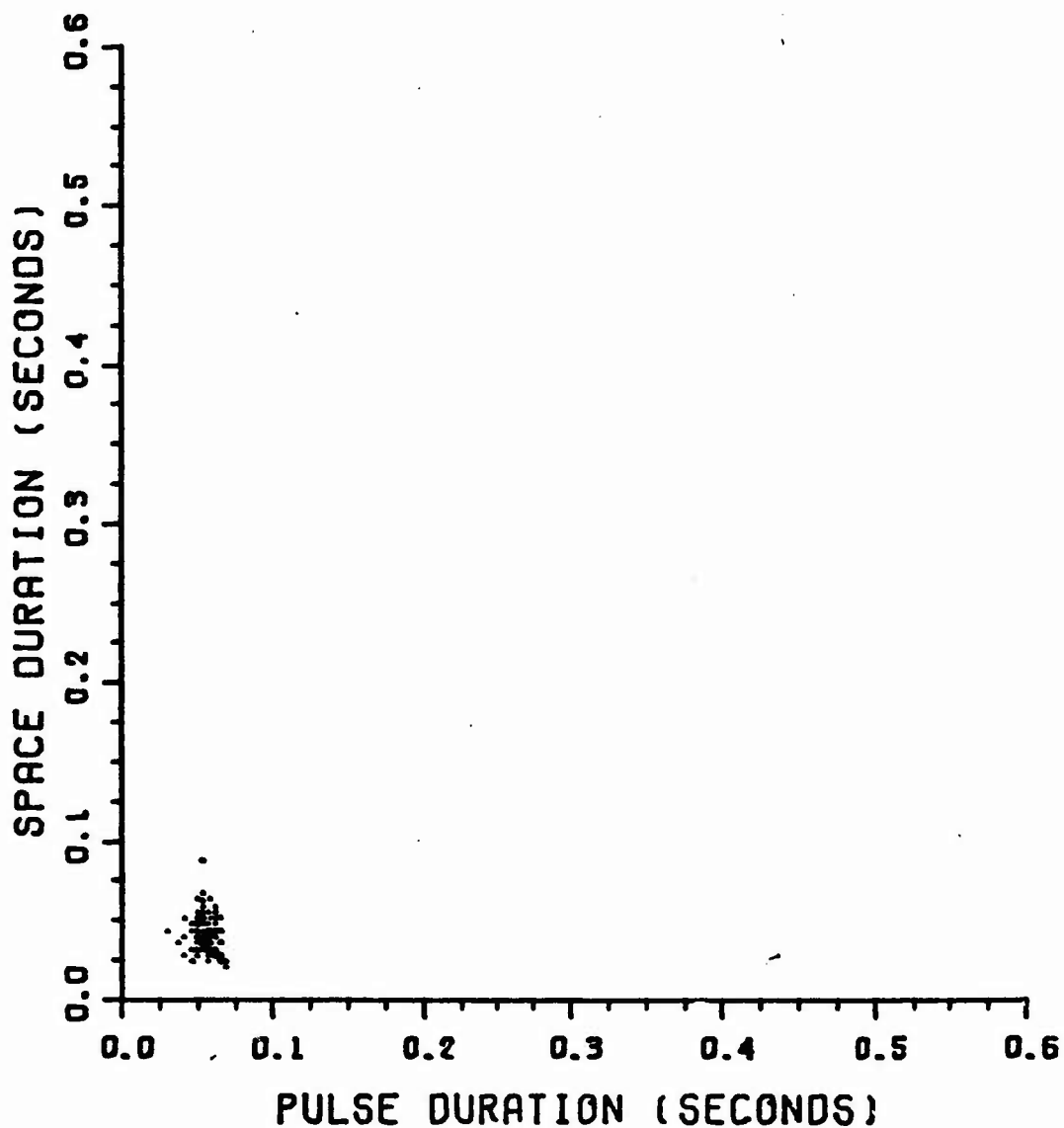


Fig. B-3. Morse Code Data Distribution Plot, DOT (Intermediate)
Time Duration vs. Time Duration of Following Space
(Recording Session 1).

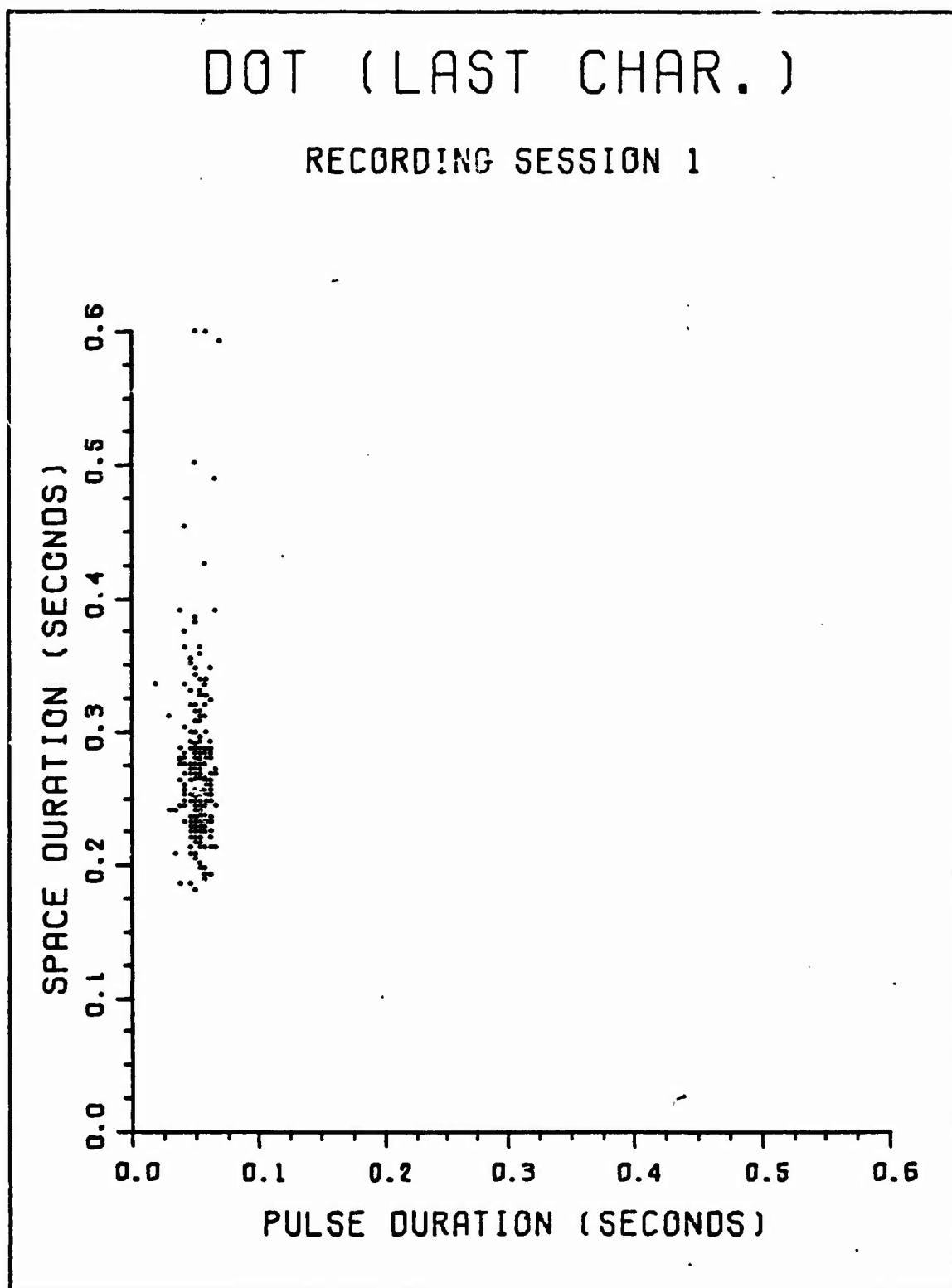


Fig. B-4. Morse Code Data Distribution Plot, DOT (Last Character)
Time Duration vs. Time Duration of Following Space
(Recording Session 1).

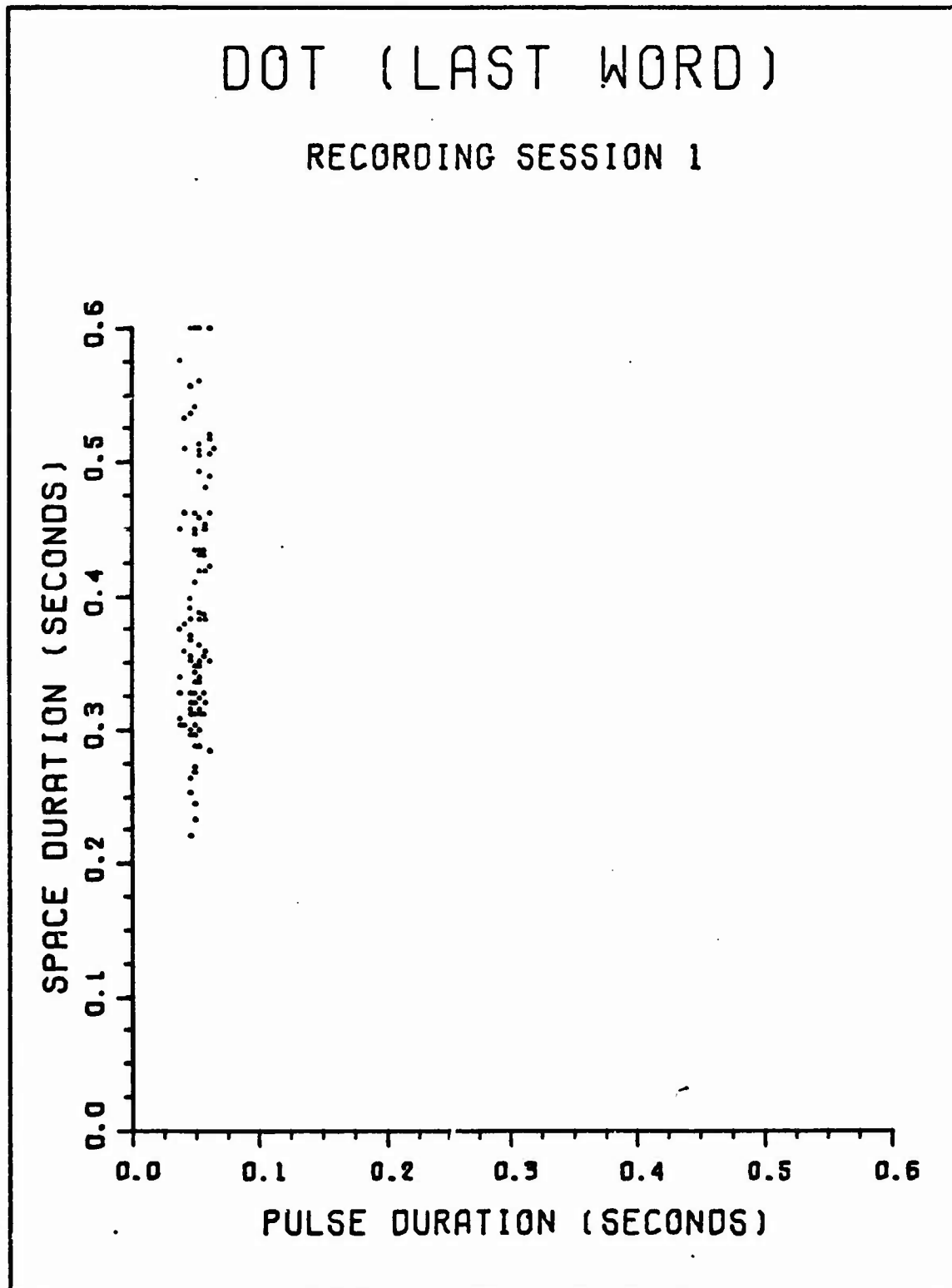


Fig. E-5. Morse Code Data Distribution Plot, DOT (Last Word)
Time Duration vs. Time Duration of Following Space
(Recording Session 1).

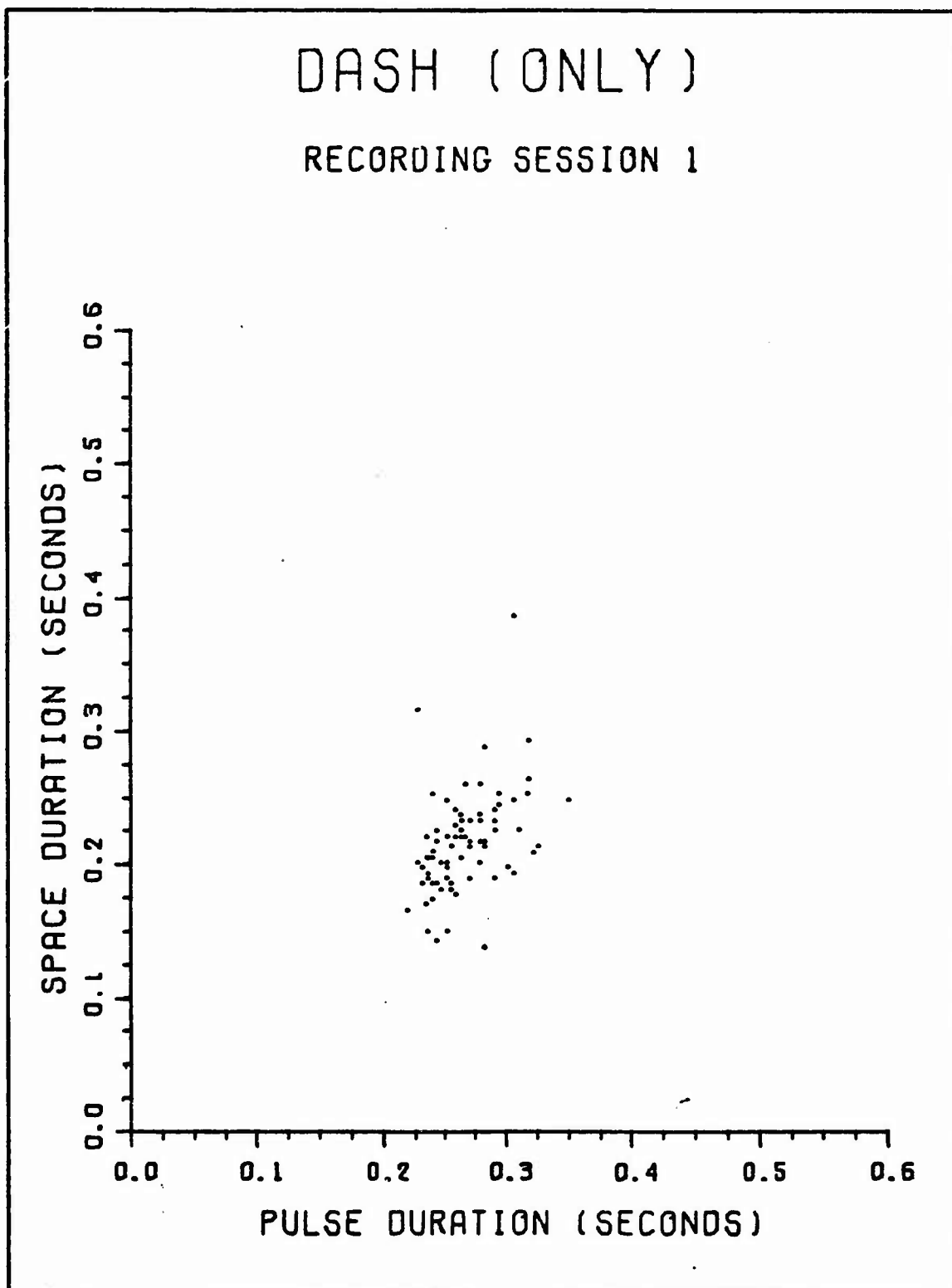


Fig. B-6. Morse Code Data Distribution Plot, DASH (Only)
Time Duration vs. Time Duration of Following Space
(Recording Session 1).

DASH (FIRST)

RECORDING SESSION 1

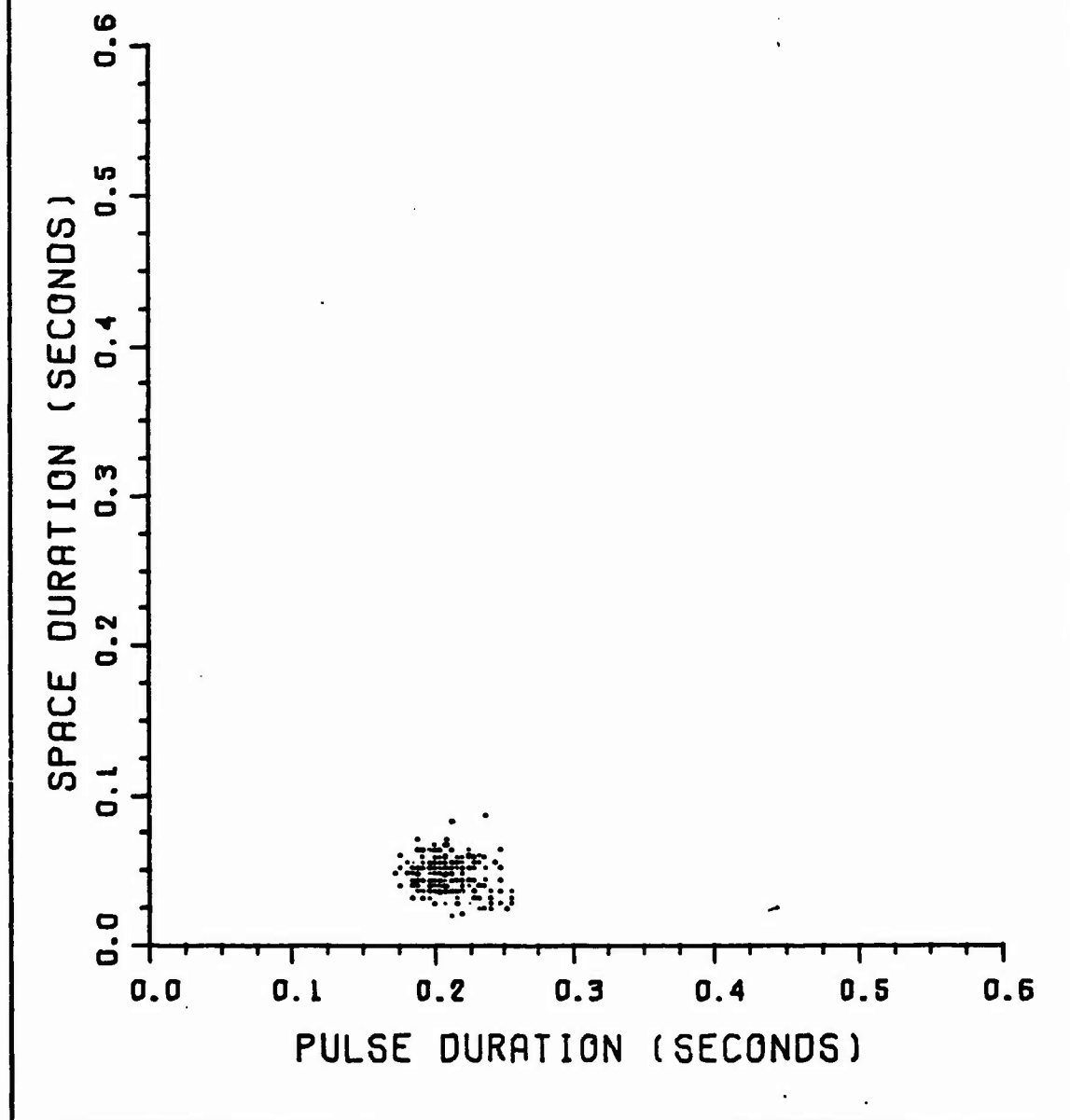


Fig. B-7. Morse Code Data Distribution Plot, DASH (First)
Time Duration vs. Time Duration of Following Space
(Recording Session 1).

DASH (INTERMEDIATE)

RECORDING SESSION 1

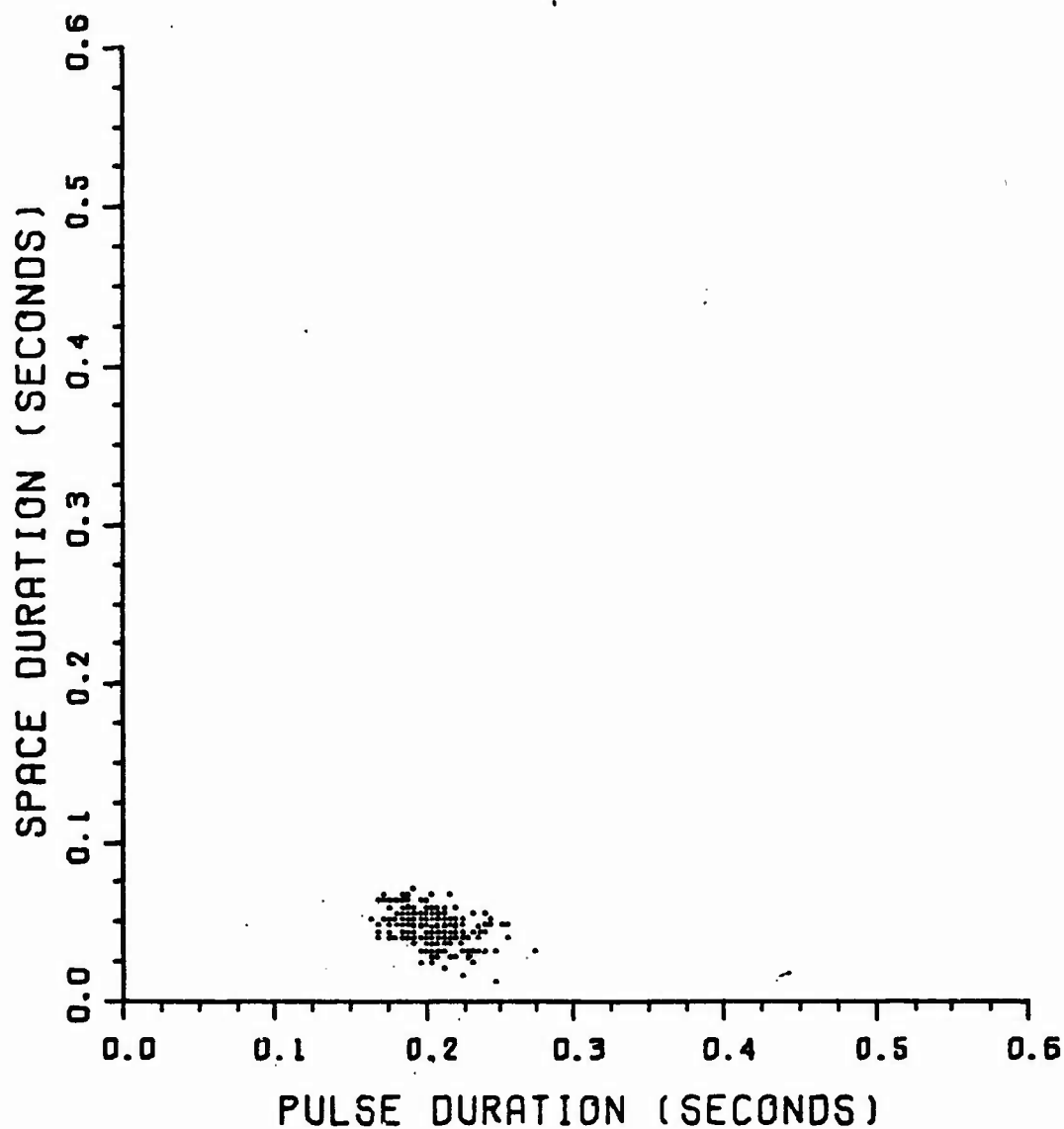


Fig. B-8. Morse Code Data Distribution Plot, DASH (Intermediate)
Time Duration vs. Time Duration of Following Space
(Recording Session 1).

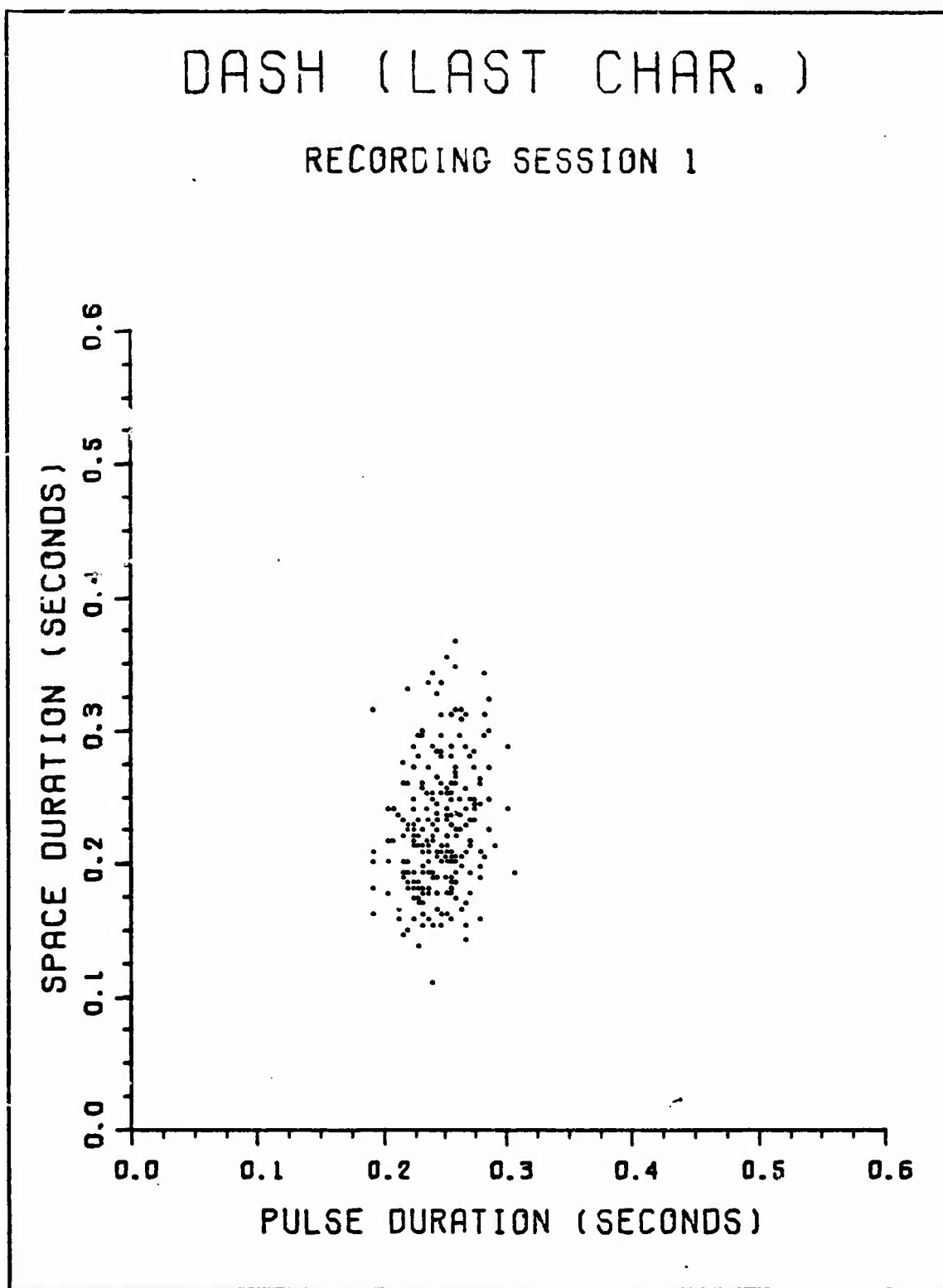


Fig. B-9. Morse Code Data Distribution Plot, DASH (Last Character)
Time Duration vs. Time Duration of Following Space
(Recording Session 1).

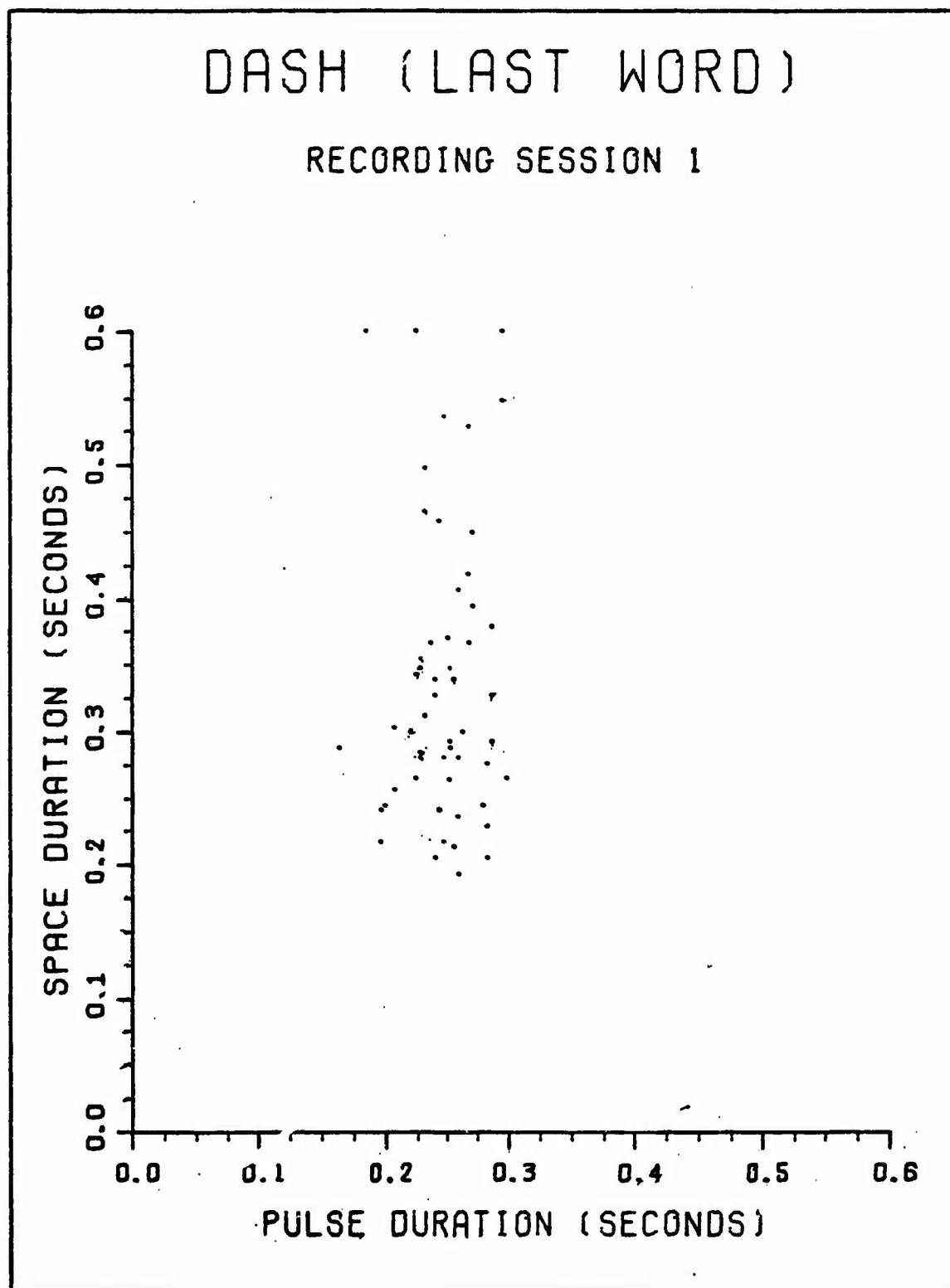


Fig. B-10. Morse Code Data Distribution Plot, DASH (Last Word)
Time Duration vs. Time Duration of Following Space
(Recording Session 1).

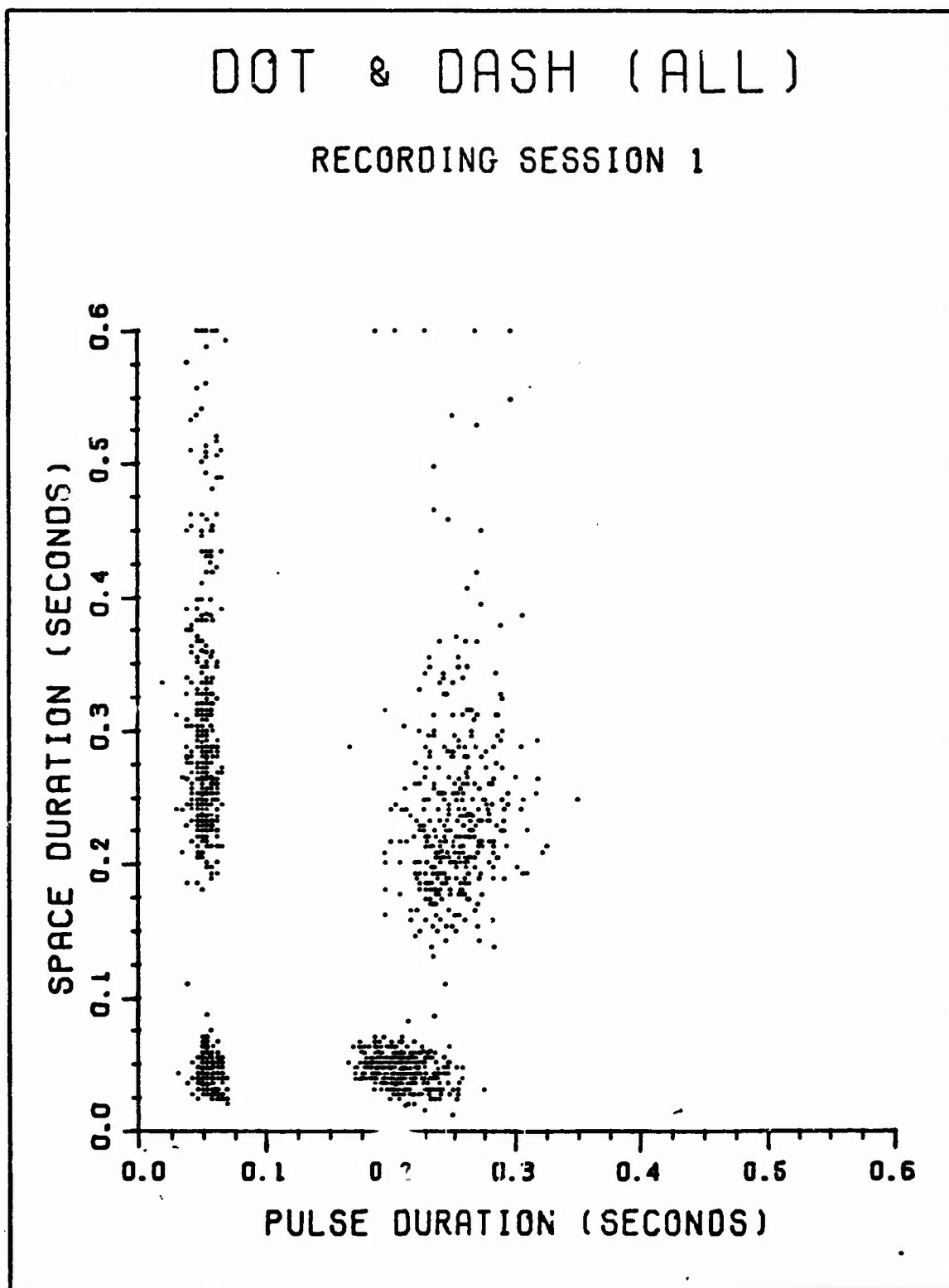


Fig. B-11. Morse Code Data Distribution Plot, Pulse Time Duration vs. Time Duration of Following Space (Recording Session 1).

DOT & DASH (ALL)

RECORDING SESSION 2

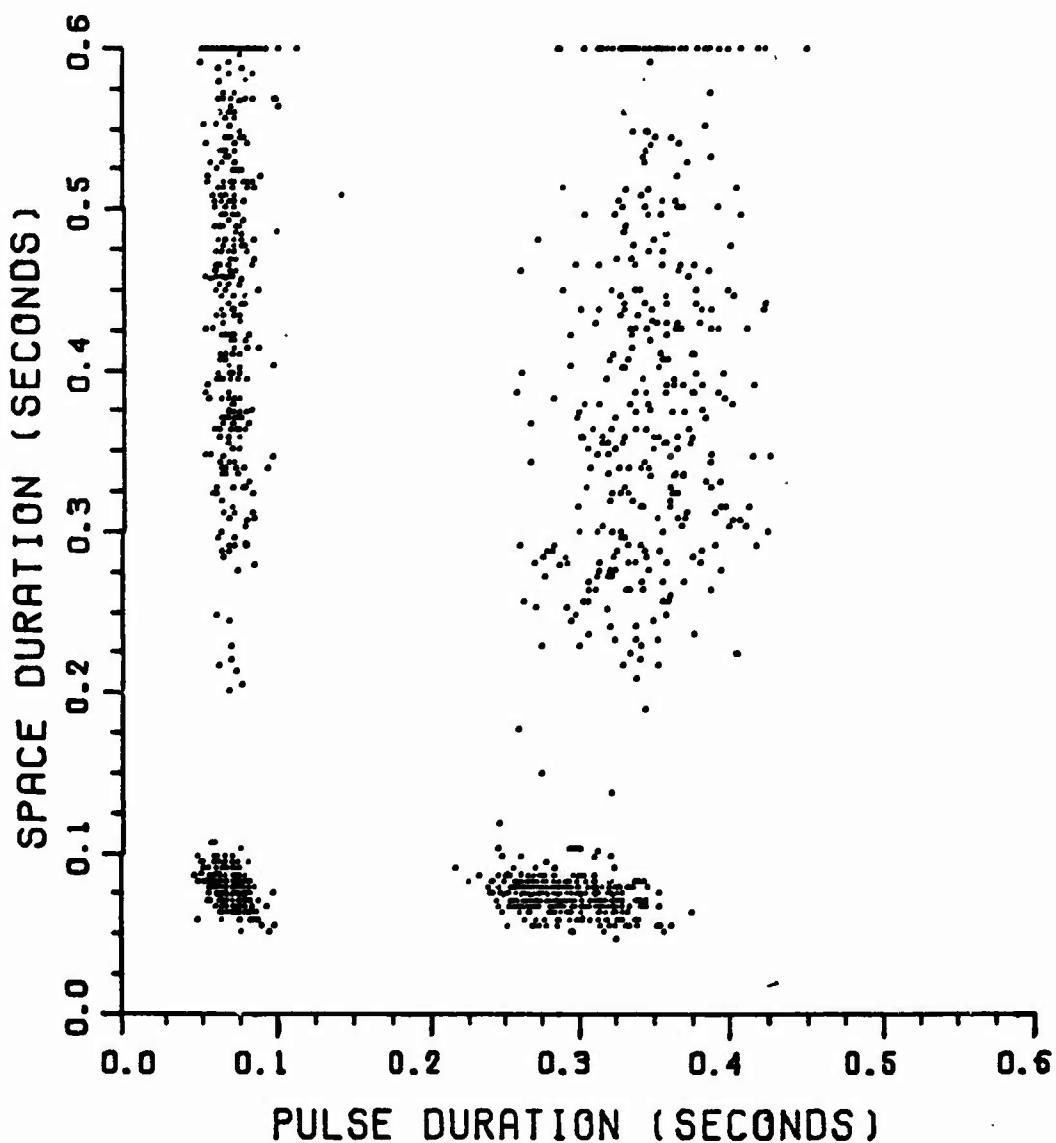


Fig. B-12. Morse Code Data Distribution Plot, Pulse Time Duration vs. Time Duration of Following Space (Recording Session 2).

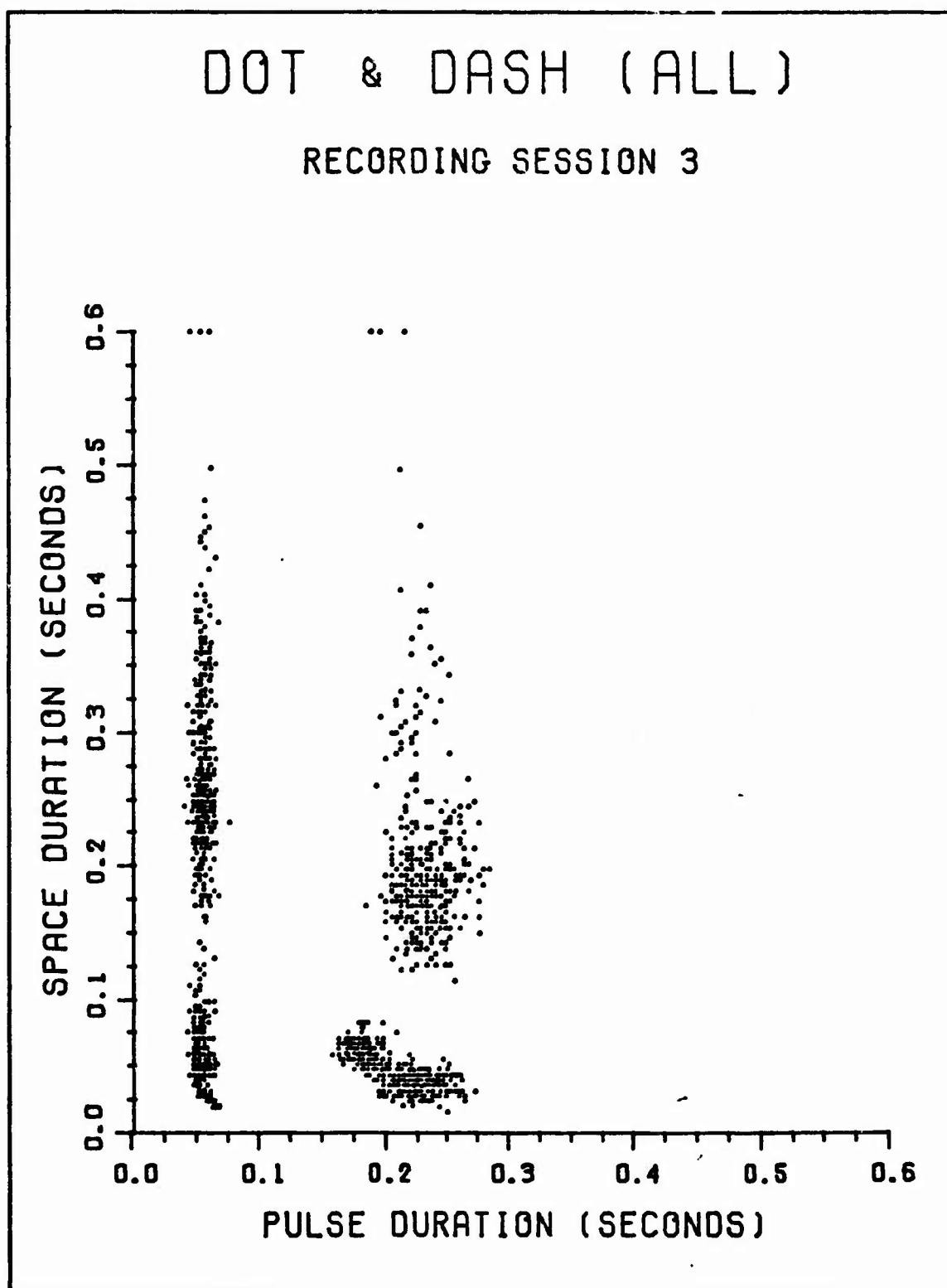


Fig. B-13. Morse Code Data Distribution Plot, Pulse Time Duration vs. Time Duration of Following Space (Recording Session 3).

DASH (LAST CHAR.)

RECORDING SESSION 1

X = PULSE Y = SPACE

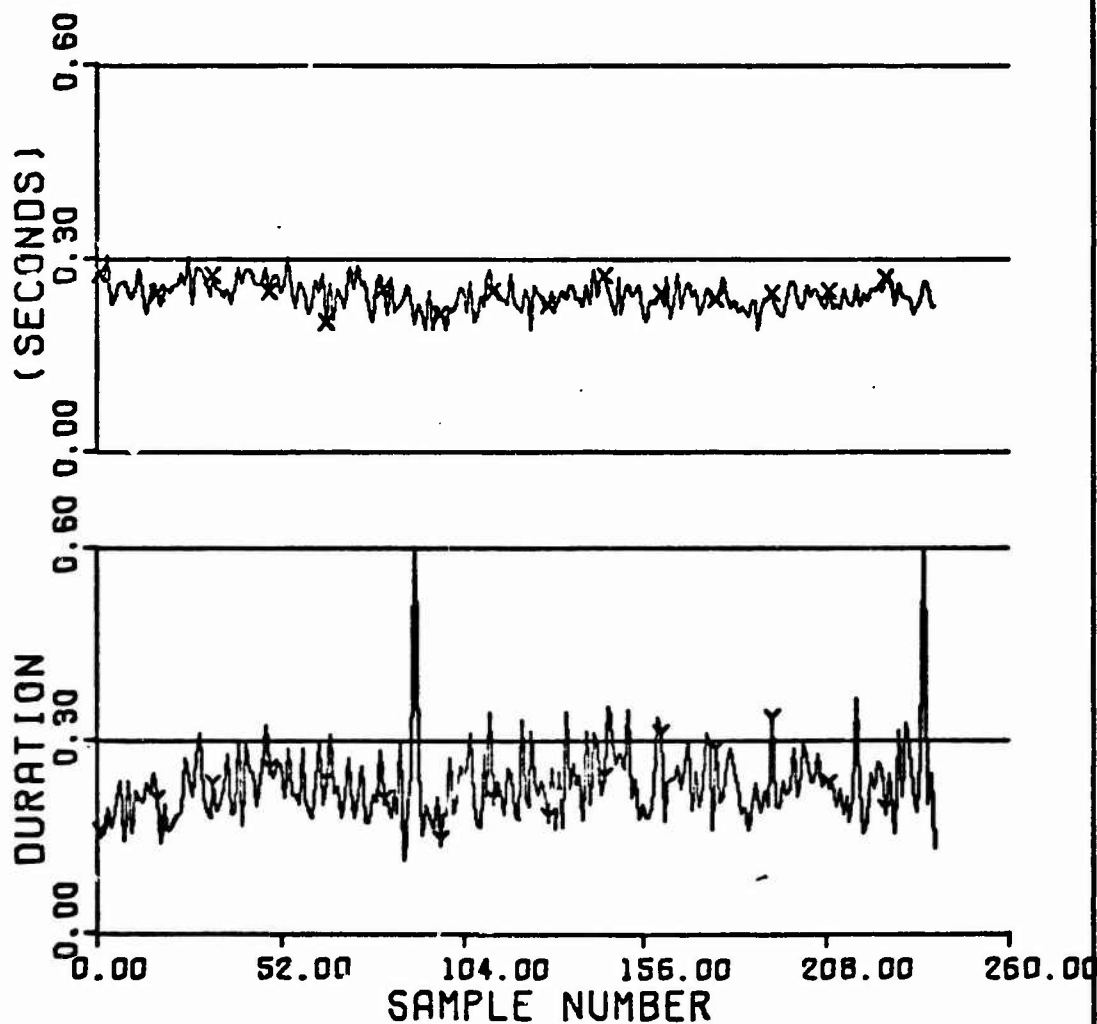


Fig. B-14. DASH (Last Character) and Following Space
Time Duration Fluctuations (Recording Session 1).

DASH (LAST CHAR.)

RECORDING SESSION 2

X = PULSE Y = SPACE

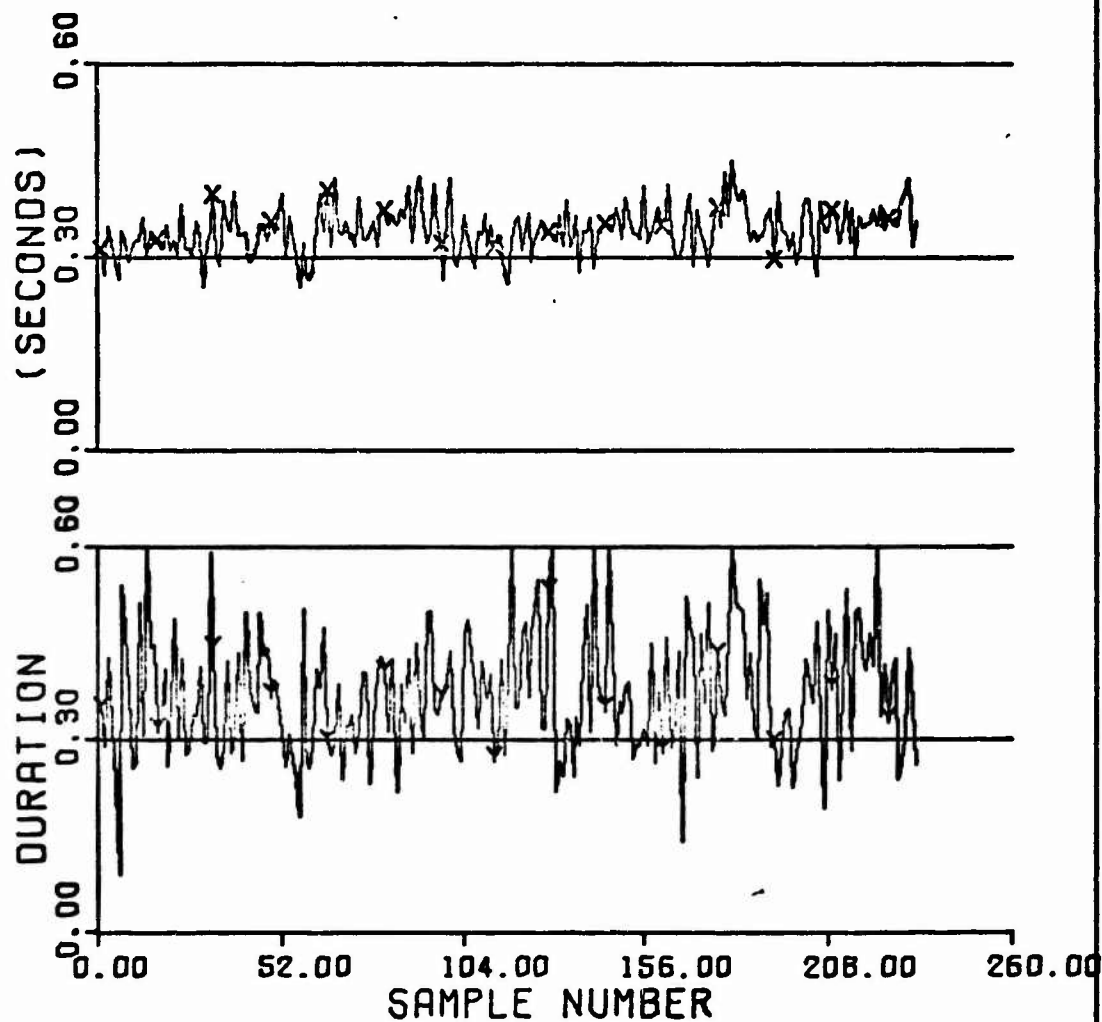


Fig. B-15. DASH (Last Character) and Following Space
Time Duration Fluctuations (Recording Session 2).

Appendix C

Code and Character Listing

This appendix contains a list of 47 international Morse code characters and code representations. Corresponding internal code words and Teletypewriter output characters used in the recognition program are also listed.

Letters

<u>Morse Character</u>	<u>Morse Code</u>	<u>Internal Code</u>	<u>Printer Character</u>
A	. -	2002	A
B	- . . .	4004	B
C	- . - .	5004	C
D	- . .	4003	D
E	.	0001	E
F	. . - .	1004	F
G	- - .	6003	G
H		0004	H
I	. .	0002	I
J	. - - -	3404	J
K	- . -	5003	K
L	. - . .	2004	L
M	- -	6002	M
N	- .	4002	N
O	- - -	7003	O
P	. - - .	3004	P
Q	- - . -	6404	Q
R	. - .	2003	R
S	. . .	0003	S
T	-	4001	T
U	. . -	1003	U
V	. . . -	0404	V
W	. - -	3003	W
X	- . . -	4404	X

Letters Cont.

<u>Morse Character</u>	<u>Morse Code</u>	<u>Internal Code</u>	<u>Printer Character</u>
Y	- . - -	5404	Y
Z	- - . .	6004	Z

Numbers

0	- - - - -	7605	0
1	. - - - -	3605	1
2	. . - - -	1605	2
3	. . . - -	0605	3
4	 -	0205	4
5		0005	5
6	-	4005	6
7	- - . . .	6005	7
8	- - - . .	7005	8
9	- - - - .	7405	9

Punctuations and Special Functions

Period	. - . - . -	2506	.
Comma	- - . . - -	6306	,
Question Mark	. . - - . .	1406	?
Colon	- - - . . .	7006	:
Semicolon	- . - . - .	5206	;
Double Dash	- . . . -	4205	-
Parenthesis	- . - - . -	5506	)

Punctuations and Special Functions Cont.

<u>Morse Character</u>	<u>Morse Code</u>	<u>Internal Code</u>	<u>Printer Character</u>
Fraction Bar	- . . - .	4450	/
Error ^a		0010	<
Wait	. - . . .	2005	\
End of Message	. - . - .	2405	*
End of Work	. . . - . -	0506	\$
Space ^b	(none)	0000	(Space)
Unknown ^b	?	?	=

a Usually a string of 8 DOTs. May be 6 or more.

b Unique to Recognition Program.

Appendix D

Computer Test Messages

This appendix contains recognition program outputs for Recording Sessions 1 through 7. Copies of the text used to transmit Recording Sessions 1 through 4 and Recording Session 6 are also presented. Output errors (discrepancies between the text and the output) are indicated by astericks (*) and number symbols (#) located below the error. Astericks indicate errors made by the message sender; number symbols indicate errors made by the recognition program. See Table IV (Chapter VI) for recording session statistics and error percentages.

AMATEUR RADIO IS A SCIENTIFIC HOBBY, A MEANS OF GAINING PERSONAL SKILL IN THE FASCINATING ART OF ELECTRONICS AND AN OPPORTUNITY TO COMMUNICATE WITH FELLOW CITIZENS BY PRIVATE SHORT WAVE RADIO. SCATTERED OVER THE GLOBE ARE OVER 350,000 AMATEUR RADIO OPERATORS WHO PERFORM A SERVICE DEFINED IN INTERNATIONAL LAW AS ONE OF SELF TRAINING, INTERCOMMUNICATION AND TECHNICAL INVESTIGATIONS CARRIED ON BY DULY AUTHORIZED PERSONS INTERESTED IN RADIO TECHNIQUE SOLELY WITH A PERSONAL AIM AND WITHOUT PECUNIARY INTEREST. FROM A HUMBLE BEGINNING AT THE TURN OF THE CENTURY, AMATEUR RADIO HAS GROWN TO BECOME AN ESTABLISHED INSTITUTION. TODAY THE AMERICAN FOLLOWERS OF AMATEUR RADIO NUMBER OVER 250,000, TRAINED COMMUNICATORS FROM WHOSE RANKS WILL COME THE PROFESSIONAL COMMUNICATIONS SPECIALISTS AND EXECUTIVES OF TOMORROW - JUST AS MANY OF TODAY'S RADIO LEADERS WERE FIRST ATTRACTED TO RADIO BY THEIR EARLY INTEREST IN AMATEUR RADIO COMMUNICATION. A POWERFUL AND PROSPEROUS ORGANIZATION NOW PROVIDES A BOND BETWEEN AMATEURS AND PROTECTS THEIR INTERESTS. AN INTERNATIONALLY RESPECTED MAGAZINE IS PUBLISHED SOLELY FOR THEIR BENEFIT. THE MILITARY SERVICES SEEK THE COOPERATION OF THE AMATEUR IN DEVELOPING COMMUNICATIONS RESERVES. AMATEUR RADIO SUPPORTS A MANUFACTURING INDUSTRY WHICH, BY THE VERY DEMANDS OF AMATEURS FOR THE LATEST AND BEST EQUIPMENT, IS ALWAYS UP TO DATE IN ITS DESIGNS AND PRODUCTION TECHNIQUES - IN ITSELF A NATIONAL ASSET. AMATEURS HAVE WON THE GRATITUDE OF THE NATION FOR THEIR HEROIC PERFORMANCES IN TIMES OF NATURAL DISASTER. TRADITIONAL AMATEUR SKILLS IN EMERGENCY COMMUNICATION ARE ALSO THE STAND BY SYSTEM FOR THE NATION'S CIVIL DEFENSE. AMATEUR RADIO IS, INDEED, A MAGNIFICENTLY USEFUL INSTITUTION.

(Sheet 1 of 2)

Fig. D-1. Prepared Text for Recording Sessions 1 through 4.

ALTHOUGH AS OLD AS THE ART OF RADIO ITSELF, AMATEUR RADIO DID NOT ALWAYS ENJOY SUCH PRESTIGE. ITS FIRST ENTHUSIASTS WERE PRIVATE CITIZENS OF AN EXPERIMENTAL TURN OF MIND WHOSE IMAGINATIONS WENT WILD WHEN MARCONI FIRST PROVED THAT MESSAGES ACTUALLY COULD BE SENT BY WIRELESS. THEY SET ABOUT LEARNING ENOUGH ABOUT THE NEW SCIENTIFIC MARVEL TO BUILD HOMEMADE SPARK TRANSMITTERS. BY 1912 THERE WERE NUMEROUS GOVERNMENT AND COMMERCIAL STATIONS, AND HUNDREDS OF AMATEURS. REGULATION WAS NEEDED, SO LAWS, LICENSES AND WAVELENGTH SPECIFICATIONS APPEARED. THERE WAS THEN NO AMATEUR ORGANIZATION NOR SPOKESMAN. BUT AS THE YEARS ROLLED ON, AMATEURS FOUND OUT HOW, AND DX JUMPED FROM LOCAL TO 500 MILE AND EVEN OCCASIONAL 1000 MILE TWO WAY CONTACTS. BECAUSE ALL LONG DISTANCE MESSAGES HAD TO BE RELAYED, RELAYING DEVELOPED INTO A FINE ART - AN ABILITY THAT WAS TO PROVE INVALUABLE WHEN THE GOVERNMENT SUDDENLY CALLED HUNDREDS OF SKILLED AMATEURS INTO WAR SERVICE IN 1917. MEANWHILE U.S. AMATEURS BEGAN TO WONDER IF THERE WERE AMATEURS IN OTHER COUNTRIES ACROSS THE SEAS AND IF, SOME DAY, WE MIGHT NOT SPAN THE ATLANTIC ON 200 METERS. MOST IMPORTANT OF ALL, THIS PERIOD WITNESSED THE BIRTH OF THE AMERICAN RADIO RELAY LEAGUE, THE AMATEUR RADIO ORGANIZATION WHOSE NAME WAS TO BE VIRTUALLY SYNONYMOUS WITH SUBSEQUENT AMATEUR PROGRESS AND SHORT WAVE DEVELOPMENT. CONCEIVED AND FORMED BY THE FAMOUS INVENTOR, THE LATE HIRAM PERCY MAXIM, ARRL WAS FORMALLY LAUNCHED IN EARLY 1914. IT HAD JUST BEGUN TO EXERT ITS FULL FORCE IN AMATEUR ACTIVITIES WHEN THE UNITED STATES DECLARED WAR IN 1917, AND BY THAT ACT SOUNDED THE KNELL FOR AMATEUR RADIO FOR THE NEXT TWO AND A HALF YEARS. THERE WERE THEN OVER 6000 AMATEURS. OVER 4000 OF THEM SERVED IN THE ARMED FORCES DURING THAT WAR.

(Sheet 2 of 2)

Fig. D-1. Prepared Text for Recording Sessions 1 through 4.

AMATEUR RADIO IS A SCIENTIFIC HOBBY. A MEANS OF GAINING PERSONAL SKILL
 IN THE FASCINATING ART OF ELECTRONICS AND AN OPPORTUNITY TO COMMUNICATE
 WITH FELLOW CITIZENS BY MEANS OF SHORTWAVE RADIO. SCATTERED OVER
 THE GLOBE ARE OVER 50,000 AMATEUR RADIO OPERATORS WHO PERFORM
 AS A SERVICE DEFINED IN INTERNATIONAL LAW AS ONE OF SELF TRAINING,
 INTERCOMMUNICATION AND TECHNICAL INVESTIGATIONS CARRIED ON BY
 DULY AUTHORIZED PERSONS INTERESTED IN RADIO ELECTRICITY FOR
 PERSONAL AIM AND WITHOUT PECUNIARY INTEREST. FROM A HUMBLE BEGINNING
 AT THE TURN OF THE CENTURY, AMATEUR RADIO HAS GROWN TO BECOME AN
 ESTABLISHED INSTITUTION. TODAY THE AMERICAN FOLLOWERS OF AMATEUR
 RADIO NUMBER OVER 250,000, TRAINED COMMUNICATORS FROM WHOSE RANKS
 WILL COME THE PROFESSIONAL COMMUNICATIONS SPECIALISTS AND EXECUTIVES
 OF TOMORROW - JUST AS MANY OF TODAY'S RADIO LEADERS WERE FIRST
 ATTRACTED TO RADIO BY THEIR EARLY INTEREST IN AMATEUR RADIO
 COMMUNICATION. A POWERFUL AND PROSPEROUS ORGANIZATION NOW PROVIDES
 A BOND BETWEEN AMATEURS AND PROTECTS THEIR INTERESTS. AN
 INTERNATIONAL BUT RESPECTED MAGAZINE IS PUBLISHED SOLELY FOR
 THEIR BENEFIT. THE MILITARY SERVICES SEEK THE COOPERATION OF
 THE AMATEUR IN DEVELOPING COMMUNICATIONS RESERVES. AMATEUR
 RADIO SUPPORTS MANUFACTURING INDUSTRY WHICH, BY THE
 IT ERY DEMANDS OF AMATEURS FOR THE LATEST STAND-BY EQUIPMENT,
 IS ALWAYS UP TO DATE IN ITS DESIGN AND PRODUCTION TECHNIQUES -
 IN ITSELF A NATIONAL ASSET. AMATEURS HAVE WON

(Sheet 1 of 3)

Fig. D-2. Recognition Program Output for Recording Session 1.

THE GREATITUDE OF THE NATION FOR THE HEROIC PERFORMANCES IN TIMES OF NATURAL DISASTER. TRADITIONAL AMATEUR SKILLS IN EMERGENCY COMMUNICATION ARE ALSO THE STANDARD SYSTEM FOR THE NATION'S CIVIL DEFENSE. AMATEUR RADIO IS, INDEED, A MAGNIFICENTLY USEFUL INSTITUTION. ALTHOUGH IT SOLD AS THE ART OF RADIO ITSELF, AMATEUR RADIO DID NOT ALWAYS ENJOY SUCH A PRESTIGE. ITS FIRST ENTHUSIASTS WERE PRIVATE CITIZENS OF AN EXPERIMENTAL TURN OF MIND WHOSE IMAGINATIONS WENT WILD WHEN MARCONI FIRST PROVED THAT MESSAGES ACTUALLY COULD BE SENT BY WIRELESS. THEY SET ABOUT LEARNING ENOUGH ABOUT THE NEW SCIENTIFIC MARVEL TO BUILD HOME MADE SPARK TRANSMITTERS. BY 1912 THERE WERE NUMEROUS GOVERNMENT AND COMMERCIAL STATIONS AND HUNDREDS OF AMATEURS. BUT REGULATION WAS NEEDED, SOLA WAVELENGTH SPECIFICATIONS WERE ESTABLISHED. THERE WAS THEN NO AMATEUR ORGANIZATION NOR A SPOKESMAN. BUT AS THEY GROWED UP, AMATEURS FOUND OUT HOW, AND DX JUMPED FROM LOCAL TO 500 MILE AND EVEN OCCASIONAL 1000 MILE TWO WAY CONTACTS. BECAUSE ALL LONG DISTANCE MESSAGES HAD TO BE RELAYED, RELAYING DEVELOPED INTO A FINE ART - AN ABILITY THAT WAS TO PROVE INVALUABLE WHEN THE GOVERNMENT SUDDENLY CALLED HUNDREDS OF SKILLED AMATEURS INTO WAR SERVICE IN 1917. MEANWHILE U.S. AMATEURS BEGAN TO WONDER IF THERE WERE AMATEURS IN OTHER COUNTRIES SET ACROSS THE SEAS AND IF, SOME DAY, WE MIGHT NOT SPAN THE ATLANTIC ON 200 METERS. MOST IMPORTANT OF ALL, THIS IS PERIOD WHEN IT NEEDED THE BIRTH OF THE AMERICAN

(Sheet 2 of 3)

Fig. D-2. Recognition Program Output for Recording Session 1.

AN RADIO RELAY LEAGUE, THE AMATEUR RADIO ORGANIZATION WHOSE
 NAME WAS TO BE VIRTUALLY SYNONYMOUS WITH SUBSEQUENT AMATEUR
 FROM RESEARCH AND SHORT WAVE DEVELOPMENT. CONCEIVED AND FORMED BY
 THE FAMOUS INVENTOR, THE LATE HIRAM PERCY MAXIM, ARR L WAS FORMAL
 LY LAUNCHED IN EARLY 1914. IT HAD JUST BEGUN TO EXERT ITS FULL
 FORCE IN AMATEUR ACTIVITIES WHEN THE UNITED STATES DECLARED
 WAR IN 1917, AND BY THAT ACT SOUNDED THE KNELL FOR AMATEUR
 RADIO FOR THE NEXT TWO AND A HALF YEARS. THERE WERE THEN OVER
 6000 AMATEURS. OVER 4000 OF THEM SERVED IN THE ARMED FORCES DURING
 THAT WAR.

(Sheet 3 of 3)

Fig. D-2. Recognition Program Output for Recording Session 1.

AMATEUR RADIO IS AN INTERESTING HOBBY, A MEANS OF GAINING PERSONAL SKILL IN THE FASCINATING WORLD OF ELECTRONICS AND AN OPPORTUNITY TO COMMUNICATE WITH FELLOW CITIZENS BY PRIVATE SHORT WAVE RADIO. SCATTERED OVER THE GLOBE ARE OVER 350,000 AMATEUR RADIO OPERATORS WHO PERFORM A SERVICE DEFINED IN INTERNATIONAL LAW AND SOME OF SELF TRAINING, INTERCOMMUNICATION AND TECHNICAL INVESTIGATIONS CARRIED OUT BY DULY AUTHORIZED PERSONS INTERESTED IN RADIO TECHNIQUE SOLELY WITH A PERSONAL AIM AND WITHOUT PECUNIARY INTEREST. FROM ABOUT THE BEGINNING OF THE TURN OF THE CENTURY, AMATEUR RADIO HAS GROWN TO BECOME AN ESTABLISHED INSTITUTION. TODAY THERE ARE OVER 250,000 TRAINED COMMUNICATORS FROM WHOSE RANKS WILL COME THE PROFESSIONAL COMMUNICATIONS SPECIALISTS AND EXECUTIVES OF TOMORROW - PLUS MANY OF TODAY'S RADIO LEADERS WERE FIRST ATTRACTED TO RADIO BY THEIR EARLY INTEREST IN AMATEUR RADIO COMMUNICATION. A POWERFUL AND PROSPEROUS ORGANIZATION NOW PROVIDES AID BETWEEN AMATEURS AND PROTECTS THEIR INTERESTS. AN INTERNATIONALLY RESPECTED MAGAZINE IS PUBLISHED SOLELY FOR THEIR BENEFIT. THE MILITARY SERVICES SEEK THE COOPERATION OF THE AMATEUR IN DEVELOPING COMMUNICATIONS RESERVES. AMATEUR RADIO SUPPORTS A MANUFACTURING INDUSTRY WHICH, BY THE VERY DEMANDS OF AMATEURS FOR THE LATEST AND BEST EQUIPMENT, IS ALWAYS UP TO DATE IN ITS DESIGNS AND PRODUCTION TECHNIQUES - IN ITSELF A NATIONAL

(Sheet 1 of 3)

Fig. D-3. Recognition Program Output for Recording Session 2.

TIONAL ASS E T . AMATEURS HAVE WON THE GRATITUDE OF THE NATION F
 OR THEIR HEROIC P ERFORMANC E S IN TIME S OF NATURAL D ISAS TER.
 TRADITIONAL AMATEUR SKILL S IN EMERG ENC Y COMMUNICATION ARE ALS
 O THE STAND BY SY STEM FOR THE NATIONS CIVIL DEF ENS E. AMA TEUR
 RADIOIS, INDEED , A MAGNIF IC ENTLY USEFUL INSTITUTION . ALTHOUG
 H AS OLD A S THE ARTOF RADIO ITS ELF , AMATEUR RADIO DID NOT ALW
 AYS ENJOY SUCH PRESTIG E. ITSE IR ST ENTHUSIASTS WER E PR IVATE
 C ITIZENS OF AN EXPERIMEN TALTURN OF MIND WHOS E IMAGINATIONS WE
 NT W IL D WHEN MARCON I FIRST PROVED TH AT MESSAG ES ACTUALLY COU
 LD BE SENT BY WIREL ESSW . TH EY S E T AB OUT LEARNING ENOUGH AB
 OUT THE NEW SCIENTIFIC MARVEL TO BUILD HOMEMAD E SPARK TRANSM IT
 TERS. BY17 9 1 2 TH ERE WERE NUMEROUS G O VER NM ENT AND C OMM E
 RCIAL STATIONS, AND HUNDR EDS OF AMATEURS. REGULATION RA S N EED
 ED, SO LAWS , LICENSES AND WAVEL E NG T H SPEC IFICATION S APPEA
 RED. THERE WAS THEN NO AMATEUR ORGANIZATION NOR SPKESMAN. B UT
 AS THE YEARS ROLLED ON, AMATEURS FOUND OUT HOW , AND DX NUM P ED
 FROM LOC AL TO5 0 0 MIL I AND E VEN OCC AS IONAL 1000 M IL E TWO
 WAYCONTACTS . BECAUSE ALL LONG DISTANC E MESSAG ES HAD TO B E RE
 LAYED , R EL AYING D EVELO P ED INTO IF INE ART- AN ABIL ITY THA
 T WAS TO P R OVE I N VAL UAB L E WH E N TH E G O V E R N M E N T
 SUDDENLY CAL L EDHUND R E D S OF SKILL ED AMATEURS INTOWAR SERVI
 C IN1917 # MEANWHIL E U. S. AMATEURS BEGAN THO WONDER IF THER E
 WERE AMATE URS INOTHER COUNTRIES ACROSS THE SEAS AND IF , SOME D

(Sheet 2 of 3)

Fig. D-3. Recognition Program Output for Recording Session 2.

AY, WE MIGHT NOW SPAN THE ATLANTIC ON 200 METERS. MOST IMPORTA
 NT OF ALL, THIS PERIOD WITNESSED THE BIRTH OF THE AMERICAN
 CANADIAN AMATEUR RADIO LEAGUE, THE AMATEUR RADIO ORGANIZATION WHOSE
 NAME WAS TO BE VIRTUALLY SYNONYMOUS WITH SUBSEQUENT AMATEUR
 RADIO PROGRESS AND SHORT WAVE DEVELOPMENT. CONCEIVED AND FORMED BY
 THE FAMOUS INVENTOR, THE LATE HILARY PERCY MAXIM, ARRL
 WAS FORMALLY LAUNCHED IN EARLY 1915. IT HAD JUST BEGUN TO E
 XERT ITS FULL FORCE IN AMATEUR ACTIVITIES WHEN THE UNITED STA
 TES DECLARED WAR IN 1917, AND BY THAT ACT SOUNDED THE
 KNELL FOR AMATEUR RADIO FOR THE NEXT TWO AND A HALF YEARS. TH
 ERE WERE THEN OVER 6000 AMATEURS. OVER 4000 OF THEM SERVED IN THE
 ARMED FORCES DURING THAT WAR.

(Sheet 3 of 3)

Fig. D-3. Recognition Program Output for Recording Session 2.

AMATEUR RADIO IS A SCIENTIFIC HOBBY, A MEANS OF GAINING PERSONAL SKILL IN THE FASCINATING ART OF ELECTRONICS AND AN OPPORTUNITY TO COMMUNICATE WITH FELLOW CITIZENS BY PRIVATE SHORT WAVE RADIO. SO A T T E R E D O S T E R T H E T N L O B E A R E O S T E R 3 5 0, 0 0 0 A M A T E U R R A D I O O P E R A T O R H W H O P E R F O R M A S E R V I C E D E F I N E D I N I N T E R N A T I O N A L L A W A S O N E O F S E L F T R A I N I N G, I N T E R C O M M U N I C A T I O N A N D T E C H N I C A L I N V E S T I G A T I O N S C A R R I E D O N B Y D U L Y A U T H O R I Z E D P E R S O N S I N T E R E S T E D I N R A D I O T E C H N I Q U E S O L E L Y W I T H A P E R S O N A L A I M A N D W I T H O U T P E C U N I A R Y I N T E R E S T. F R O M A H U M B L E B E G I N N I N G A T T H E T U R N O F T H E C E N T U R Y, A M A T E U R R A D I O H A S G R O W N T O B E C O M E A N E S T A B L I S H E D I N S T I T U T I O N. T O D A Y T H E A M E R I C A N F O L L O W E R S O F A M A T E U R R A D I O N U M B E R O S T E R 1 0 5 0, 0 0 0, T R A I N E D C O M M U N I C A T O R S F R O M W H O S E R A N K S W I L L C O E T H E P R O F E S S I O N A L C O M M U N I C A T I O N S S P E C I A L I S T S A N D E X E C U T I V E S O F T O M O R R O W B U T J U S T A S M A N Y O F T O D A Y S R A D I O L E A D E R S W E R E F I R S T A T T R A C T E D T O R A D I O B Y T H E I R E A R L Y I N T E R E S T I N A M A T E U R R A D I O C O M M U N I C A T I O N. A P O W E R F U L A N D P R O S P E R O U S O R G A N I Z A T I O N N O W P R O V I D E S A B O N D B E T W E E N A M A T E U R S A N D P R O T E C T S T H E I R I N T E R E S T S. A N I N T E R N A T I O N A L L Y R E S P E C T E D M A G A Z I N E I S P U B L I S H E D S O L E L Y F O R T H E I R B E N E F I T. T H E M I L I T A R Y S E R V I C E S S E E K T H E C O O P E R A T I O N O F T H E A M A T E U R A N D E S T A B L I S H E D C O M M U N I C A T I O N S R E S E R V E S. A M A T E U R R A D I O S U P P O R T S A M A N U F A C T U R I N G I N D U S T R Y W H I C H, B Y T H E V E R Y D E M A N D S O F A M A T E U R S F O R T H E L A T E S T A N D B E S T E Q U I P M E N T, I S A L W A Y S U P T O D A T E I N I T S D E S I G N S A N D P R O D U C T I O N T E C H N I Q U E S B U T I N I T S E L F A N A T I O N A L A S S E T. A M A T E U R S H A V E W O N T H E G R A T I T U D E O F T H E N A T I O N F O R T H E I R H E R O I C P E R F O

(Sheet 1 of 3)

Fig. D-4. Recognition Program Output for Recording Session 3.

RMANCES IN TIMES OF NATURALDISASTER. TRADITIONAL AMATEUR SKILLS
 IN EMERGENCY COMMUNICATION ARE ALSO THE STANDARD BY WHICH THE NATION
 AND CIVIL DEFENSE. AMATEUR RADIO IS, IN EFFECT, A MAGNIFICENTLY USEFUL
 INSTITUTION. ALTHOUGH AS OLD AS THE ART OF RADIO ITSELF, AMATEUR RADIO
 DID NOT ALWAYS ENJOY SUCH PRESTIGE. ITS FIRST TENTHUS IAS TS WERE
 PRIVATE CITIZENS OF AN EARLY EXPERIMENTAL TURN OF MIND WHOSE IMAGI-
 NATIONS WENT WILD WHEN MARCONI FIRST PROVED THAT MESSAGES ACTUALLY
 COULD BE SENT BY WIRELESS. THEY SET ABOUT LEARNING ENOUGH ABOUT THE NEW
 SCIENTIFIC METHOD TO BUILD THEIR OWN SPARK TRANSMITTERS. BY 1910 THERE
 WERE NUMEROUS GOVERNMENT AND COMMERCIAL STATIONS, AND HUNDREDS OF
 AMATEURS. REGULATION WAS NEEDED, SO LAWS, LICENSES AND WAVELENGTH
 SPECIFICATIONS APPEARED. THERE WAS THEN NO AMATEUR ORGANIZATION NOR
 SPOKESMAN. BUT AS THE YEARS ROLLED ON, AMATEURS FOUND OUT HOW AND HOW TO JUMP
 FROM LOCAL TO 500 MILE AND EVEN OCCASIONAL 100 MILE TWO WAY CONTACTS.
 BECAUSE ALL LONG DISTANCE MESSAGE HAD TO BE RELAYED, RELAYING
 DEVELOPED INTO AN ART AND AN ABILITY THAT WAS TO PROVE INVALUABLE
 WHEN THE GOVERNMENT SUDDENLY CALLED HUNDREDS OF SKILLED AMATEURS
 INTO WAR SERVICE IN 1917. MEANWHILE U. S. AMATEURS BEGAN TO WONDER
 IF THERE WERE AMATEURS IN OTHER COUNTRIES ACROSS THE SEAS
 AND IF, SOMEDAY, WE MIGHT NOT SPAN THE ATLANTIC IN 200 METERS. MOST IM-
 PORTANT OF ALL, THIS PERIOD WITNESSED THE BIRTH OF THE AMERICAN RADIO
 RELAY LEAGUE, THE AMATEUR RADIO ORGANIZATION WHOSE NAME WAS TO BESTIR
 TRULY SYNONYMOUS WITH SUBSEQUENT AMATEUR RADIO PROGRESS AND SHORTWAVE

(Sheet 2 of 3)

Fig. D-4. Recognition Program Output for Recording Session 3.

E DEVELOPMENT. CONCEIVED AND FORMED BY THE FAMOUS INVENTOR, THE
 LATE HIRAM PERCY MAXIM, ARRL WAS FORMALLY LAUNCHED IN EARLY 1914. IT
 HAD JUST BEGUN TO EXERT ITS FULL FORCE IN AMATEUR ACTIVITIES WHEN THE
 UNITED STATES DECLARED WAR IN 1917, AND BY THAT ACT SOUNDED THE KNELL FOR
 AMATEUR RADIO FOR THE NEXT TWO AND A HALF YEARS. THERE WERE THEN OVER
 6000 AMATEURS. OF THESE, 1000 OF THEM SERVED IN THE ARMED FORCES DURING
 THAT WAR.

(Sheet 3 of 3)

Fig. D-4. Recognition Program Output for Recording Session 3.

AMATEUR RADIO IS A SCIENTIFIC HOBBY, A MEANS OF GAINING PERSONAL SKILL IN THE FASCINATING ART OF ELECTRONICS AND AN OPPORTUNITY TO COMMUNICATE WITH FELLOW CITIZENS BY PRIVATE SHORTWAVE RADIO. SCATTERED OVER THE GLOBE ARE OVER 350,000 AMATEUR RADIO OPERATORS WHO PERFORM A SERVICE DEFINED IN INTERNATIONAL LAW AS ONE OF SELF TRAINING, INTERCOMMUNICATION, AND TECHNICAL INVESTIGATIONS CARRIED ON BY DULY AUTHORIZED PERSONS INTERESTED IN RADIO TECHNIQUES SOLELY WITH A PERSONAL AIM AND WITHOUT PROFIT OR COMMERCIAL INTEREST. FROM A HUMBLE BEGINNING AT THE TURN OF THE CENTURY, AMATEUR RADIO HAS GROWN TO BECOME AN ESTABLISHED INSTITUTION. TODAY THE AMERICAN FOLLOWERS OF AMATEUR RADIO NUMBER OVER 250,000, TRAINED COMMUNICATORS FROM WHOSE RANKS WILL COME THE PROFESSIONAL COMMUNICATIONS SPECIALISTS AND EXECUTIVES OF TOMORROW - JUST AS MANY OF TODAY'S RADIO PIONEERS WERE FIRST ATTRACTED TO RADIO BY THEIR EARLY INTEREST IN AMATEUR RADIO COMMUNICATION. A POWERFUL AND PROSPEROUS ORGANIZATION NOW PROVIDES A BOND BETWEEN AMATEURS AND PROTECTS THEIR INTERESTS. AN INTERNATIONALLY RESPECTED MAGAZINE IS PUBLISHED SOLELY FOR THEIR BENEFIT. THE MILITARY SERVICES SEEK THE COOPERATION OF THE AMATEUR COMMUNICATORS IN DEVELOPING COMMUNICATIONS RESERVES. AMATEUR RADIO SUPPORTS A MANUFACTURING INDUSTRY WHICH, BY THE VERY DEMANDS OF AMATEURS FOR THE LATEST AND BEST EQUIPMENT, IS ALWAYS UP TO DATE IN ITS DESIGNS AND PRODUCTION TECHNIQUES - IN ITSELF A NATIONAL

(Sheet 1 of 3)

Fig. D-5. Recognition Program Output for Recording Session 4.

ASSET. AMATEURS HAVE WON THE GRATITUDE OF THE NATION FOR THEIR HEROIC
 PERFORMANCES IN TIMES OF NATURAL DISASTER. TRADITIONAL AMATEUR
 SKILLS IN EMERGENCY COMMUNICATION ARE ALSO THE STAND-BY SYSTEM FOR
 THE NATION'S CIVIL DEFENSE. AMATEUR RADIO IS, INDEED, A MAGNIFICENTLY
 USEFUL INSTITUTION. ALTHOUGH AS OLD AS THE ART OF RADIO ITSELF,
 AMATEUR RADIO DID NOT ALWAYS ENJOY SUCH PRESTIGE. ITS FIRST
 ENTHUSIASTS WERE PRIVATE CITIZENS OF AN EXPERIMENTAL TURN OF MIND
 WHOSE IMAGINATIONS WENT WILD WHEN MARCONI FIRST PROVED THAT
 MESSAGES ACTUALLY COULD BE SENT BY WIRELESS. THEN THEY SET
 ABOUT LEARNING ENOUGH ABOUT THE NEW SCIENTIFIC MARVEL TO BUILD
 HOME-MADE SPARK-TRANS. BY 1912 THERE WERE NUMEROUS GOVERNMENT
 AND COMMERCIAL STATIONS, AND HUNDREDS OF AMATEURS. THEREFORE
 REGULATION WAS NEEDED, SO LAWS SET LICENSES AND WAVELENGTH
 SPECIFICATIONS APPEARED. THERE WAS THEN NO AMATEUR
 ORGANIZATION NOR SPOKESMAN. BUT AS THE YEARS ROLLED ON,
 AMATEURS FOUND OUT HOW, AND DX JUMPED FROM LOCAL TO 500
 MILES AND EVEN OCCASIONAL 1000 MILE TWO-WAY CONTACTS. BECAUSE
 LONG-DISTANCE MESSAGES HAD TO BE RELAYED, RELAYING DEVELOPED INTO
 ART - PRACTICE THAT WAS TO PROVE INVALUABLE WHEN THE GOVERNMENTS
 CALLED HUNDREDS OF SKILLED AMATEURS INTO WAR SERVICE IN 1917.
 MEANWHILE U.S. AMATEURS BEGAN TO WONDER IF THERE WERE
 AMATEURS IN OTHER COUNTRIES ACROSS THE SEAS AND IF, SOME
 DAY, WE MIGHT NOT SPAN THE ATLANTIC ON 200 METERS. MOST IMPORTANT

(Sheet 2 of 3)

Fig. D-5. Recognition Program Output for Recording Session 4.

NT OF ALL, THIS PERIOD WITNESSED THE BIRTH OF THE AMERICAN RADIO RE
 LAY LEAGUE, THE AMATEUR RADIO ORGANIZATION WHOSE NAME WAS TO B
 E VIRTUALLY SYNONYMOUS WITH SUBSEQUENT AMATEUR PROGRESS AND S - AND
 S HORT-TERM DEVELOPMENT. K CONCEIVED AND FORMED BY THE FAMOUS INVENTOR
 , THE LATE HIRSH PERCY MAXIM, ARRL WAS FORMALLY LAUNCHED IN EARLY 1914. IT HAD
 ALREADY BEGUN TO EXERT ITS FULL FORCE IN AMATEUR ACTIVITIES WHEN THE UNI
 TE STATES DECLARED WAR ON GREAT BRITAIN IN 1917, AND BY
 THAT ACT SOUNDED THE KNELL FOR AMATEUR RADIO FOR THE NEXT
 TWO AND A HALF YEARS. THERE WERE THEN OVER 600 AMATEUR
 STATIONS. OVER 4000 OF THEM SERVED IN THE ARMED FORCES DURING
 THAT WAR.

(Sheet 3 of 3)

Fig. D-5. Recognition Program Output for Recording Session 4.

KUIEW9121SARXMDIKKHNUMJQU45094
 JH2B161408MAIWNIBZOECTTJ034839PKR
 C SERP OIMNEH0ZUPTCCXPFGCONAY35618
 VAHQBBZQQF03725VYYBX09662RTE

24849IDLKRLNWQBEIHCD42253IAUBT
 GLR033483MBNRATMOZIO3208UCIEUJZY
 WIEJVWDYZBAB48516GZOTYASVYFQCKUXK
 GQNSYJXPXSCMHCDGPFHPH5247141550*

GOOFYPYGM YCOGITATESQUIZZICALLY
 JONQUILSGLORIFYKINGLYPACIFICYA=
 PACIFICYACHTS

ITISAGOODIDEATOWRAPTHEBALUNWIT
 H-VINYLELECTRICALTAPFORITSEENTIRE
 LENGTHTOPROTECTTHEFINEWIRESOF THE
 SHIELD.

SMOOTHITOUTANDTAPEITINPOSITION
 EVERYFOOT.

(Sheet 1 of 2)

Fig. D-6. Recognition Program Output for Recording Session 5.

SMOOTH IT OUT AND TAPE IT IN POSITION EVERY FOOT .

... ? / - , . / - ? . . / , , - / . ? . , *

P Q 9 5 4 0 0 X 2 3 W J V 4 1 Y 9 U B C K 1 7 6 D 9 L 4 P V
1 3 2 R 5 F 3 M 0 J *

BEST RESULTS WILL BE OBTAINED WHEN THE ANTENNA IS FORTY OR FIFTY FEET HIGH AND WELL IN THE CLEAR.

WE DISLIKE TO EXCHANGE JOBSLOTS OF SIZES VARYING FROM A QUARTER UP.

DIMENSIONS OF THE ANTENNAS SUITABLE FOR THE AMATEUR BANDS ARE GIVEN IN FIGURE 3.

THE 80 METER ANTENNAS SHOULD BE MADE OF HARD DRAWN COPPER WIRE TO PREVENT STRETCHING AND SAGGING IN HEAVY WINDS.

(Sheet 2 of 2)

Fig. D-6. Recognition Program Output for Recording Session 5.

QST DE WIAW WIAW WIAW OB 450, OSCAR 309 AND APT 388 FOLLOW QST QST QST
DE WIAW WIAW WIAW QST QST QST DE WIAW WIAW WIAW OB 450, OSCAR 309 AND
APT 388 FOLLOW QST DE WIAW HR OFFICIAL BULLETIN NR 450 FROM ARRL
HEADQUARTERS CK 92 NEWINGTON CT OCTOBER 25, 1973 TO ALL RADIO AMATEURS -
NOVEMBER PRESENTS AN EXCELLENT OPPORTUNITY FOR EVERY AMATEUR TO TEST
HIS FREQUENCY MEASURING SKILLS BY TAKING PART IN AN ARRL FREQUENCY
MEASURING TEST. WIAW WILL TRANSMIT SIGNALS FOR MEASUREMENT ON NOVEMBER
10 AT 0230 AND 0530 GMT. THIS WILL BE THE EVENING OF NOVEMBER 9 AT
2130 EST ON APPROXIMATELY 3527 7078 AND 14079 KHZ. A SECOND SERIES OF
TEST SIGNALS WILL BE TRANSMITTED THREE HOURS LATER ON ABOUT 3563 7083
AND 14072 KHZ. FULL DETAILS ON HOW TO PARTICIPATE APPEAR ON PAGE 110
OF OCTOBER QST.

Fig. D-7. Recording Session 6 Text (WIAW Bulletin).

QST DE W1AW W1AW DB 450. OSCAR 309 AND APT 388 FOLLOWS
 KST QST QST DE W1AW W1AW WAWA E TIT QST QST DE E T1AW W1AW W1AW
 DB 450. OSCAR 309 AND APT 388 =THOW ES EIE QST DE W1AW HR OFFICI
 AL BULLETIN NR 450 FROM ARR=HEADQUARTERS ECK 92 NEWINGTON CT OCT
 OBER 25. 1973 TO ALL RADIO AMATEURS - NOVEMBER PRESENTS AN EXCEL
 LENT O=NORTON= FOR EVERYAMATEUR TO TEST BIL FREQUENCY MEASURING
 SKILLS BY TAKING PART IT ES NRL FREQUEND M MEASURING TEST. W1AR
 TILL TRANSMIT SIGNAMRE FOWNVSUREMENT ON NO=BER 10 AT 0230 MNDE05
 NO GMT. THIS W?L BE THEETVENING OF NOVEMBIR 9 AT 2130 EST ON APP
 ROXIMATELY 3527 7078 AND 14079 KHZ= II E IEEOND SEI OF TEST SIGN
 ALS WILL BE TRANSMITTED THREE HOURS LATER ON ABOUT 3563 7083 AND
 14072 KHZ. FULL DEAR=OWN= PARTICIPATE U=PVR ON PAGE 110E0FE0CTOBE
 LPTST.

Fig. D-8. Recognition Program Output for Recording Session 6.

(8 wpm)

2 4 8 4 9 I D L K R L N W Q B E I H C D 4 22 5 3 I A U B X T G L
 R O 3 3 4 3 8 M B N R A T M O Z I 0 3 2 0 8 U C I E U J Z Y W I
 E J V W D Y Z B A B 4 8 516 G Z O T Y A S V F Q C K U X K G Q N S V J P
 X E S C M H C D G P F H P H 5 2 4 7 1 415 5 0 F K N P Y E M R X T 1
 2 6 9 1 W V Q O J S U L D L 7 2 6 2 8 9 F J K H E G R Q N N X Z
 W O S T P V A U U C A F Y 0 7 6 1 D B G L Y E Z H K X J W R R
 Q N C G V B D L X T M O F N S Y E Q Z B P E H J U A K I W O H D 1
 3 0 7 6 Q Z K V S R C W M E P T X U L

(10 wpm)

E V H M X 11 3 4 0 F W J B W W 25 659 I L B Z A Y E N T E E Q C S P J G Q G K R P
 P O K 25309 41687 11240 3278524996 B H F D Y W C U J E I E Q R X A T Q I Z O T C J
 S D E H L O K M G N F R U C Z B Y V M A C X D W G S E F X T P R Q I N U H V J K Y A L Z B N L 0 8
 576 05327 D I O V S Y U S E K Z M F R W A H N Q X 1498 6 G J P T B 09612 C O M I D Q F P J E
 L N H R K G S B V U L T Z A M W H E W Y P Y J X E K I S G T 3478 5 X V O Z A 10945

(12 wpm)

R Q Z J T S R A I 21369 D U O C G E U N D F 02589 C T P B H 30258 F N M E D G
 X L F E 10478 H Y K G C I Z J H B 41047 J Z I J A K Y H I Z 52136 L X G K Y M W F L X 85203 N V
 E M W O U D N V 63025 P T C O U Q S B P T 74114 R Q A Q S S R A Q T P B S R U O C T P V I V D U 96
 302 W M E V N X L F W M Y G K Y L Z J H Y R A I J Z J B H I Z I C G K Y H 88521 79630 D F L Y G E
 D M W F F E N V E G C O U D H B P T C I A Q S B 97410 J Z R Q A K Y S R A 69741 L X T P B 58852
 M W U O C N V W

Fig. D-9. Recognition Program Output for Recording Session 7.

VITA

Joel Arthur Guenther was born on [REDACTED], [REDACTED]. He graduated from high school in [REDACTED] and attended Newark College of Engineering from which he received the degree of Bachelor of Science in Electrical Engineering and a commission in the U.S. Air Force in 1965. He served as a Deputy Missile Combat Crew Commander and Instructor in the Minuteman II Weapon System, Malmstrom Air Force Base, from 1966 to 1969. He was then assigned as Program Manager for Minuteman I operational testing at Vandenberg Air Force Base. He entered the Air Force Institute of Technology in June 1972.

Permanent address: [REDACTED]
[REDACTED]

This thesis was typed by [REDACTED]